# CoAtFormer: Vision Transformer with Composite Attention

**Zhiyong Chang**[1] , **Mingjun Yin**[2] , **Yan Wang**[3]
[1]Peking University
[2]The University of Melbourne
[3]Zuoyebang
changzy@pku.org.cn, mingjuny1@student.unimelb.edu.au wangyan11@zuoyebang.com

## Abstract

Transformer has recently gained significant attention and achieved state-of-the-art performance in various computer vision applications, including image classification, instance segmentation, and object detection. However, the self-attention mechanism underlying the transformer leads to quadratic computational cost with respect to image size, limiting its widespread adoption in state-of-the-art vision backbones. In this paper we introduce an efficient and effective attention module we call **Co**mposite **At**tention. It features parallel branches, enabling the modeling of various global dependencies. In each composite attention module, one branch employs a dynamic channel attention module to capture global channel dependencies, while the other branch utilizes an efficient spatial attention module to extract long-range spatial interactions. In addition, we effectively blending composite attention module with convolutions, and accordingly develop a simple hierarchical vision backbone, dubbed CoAtFormer, by simply repeating the basic building block over multiple stages. Extensive experiments show our CoAtFormer achieves state-of-the-art results on various different tasks. Without any pre-training and extra data, CoAtFormer-Tiny, CoAtFormer-Small, and CoAtFormer-Base achieve **84.4%**, **85.3%**, and **85.9%** top-1 accuracy on ImageNet-1K with **24M**, **37M**, and **73M** parameters, respectively. Furthermore, CoAtFormer also consistently outperform prior work in other vision tasks such as object detection, instance segmentation, and semantic segmentation. When further pretraining on the larger dataset ImageNet-22k, we achieve **88.7%** Top-1 accuracy on ImageNet-1K.

## 1 Introduction

In the past years, Convolution Neural Networks (CNNs) have become a defacto choice for a wide variety of computer vision tasks [Simonyan and Zisserman, 2014; He *et al.*, 2015; Ren *et al.*, 2015; He *et al.*, 2017a] since AlexNet [Krizhevsky *et al.*, 2012]. However, convolution operations merely capture local dependencies of pixels, which neglect the dependency modeling between distant pixels to some extent [Wang *et al.*, 2017]. Recently, self-attention based Transformers [Vaswani *et al.*, 2017] have achieved excellent performance in Natural Language Processing (NLP) benchmarks and became the dominant architecture for various applications. Meanwhile, numerous researchers attempt to introduce the Transformer-based architectures into vision domains, and attain promising performance in various tasks such as image classification [Dosovitskiy *et al.*, 2020; Touvron *et al.*, 2020], object detection [?; Zhu *et al.*, 2020], and semantic segmentation [Zheng *et al.*, 2020]. Vision Transformer (ViT) [Dosovitskiy *et al.*, 2020], which utilises a sequence of embedded image patches as input to stacked standard Transformer encoders, is the first fully Transformer-based architecture that

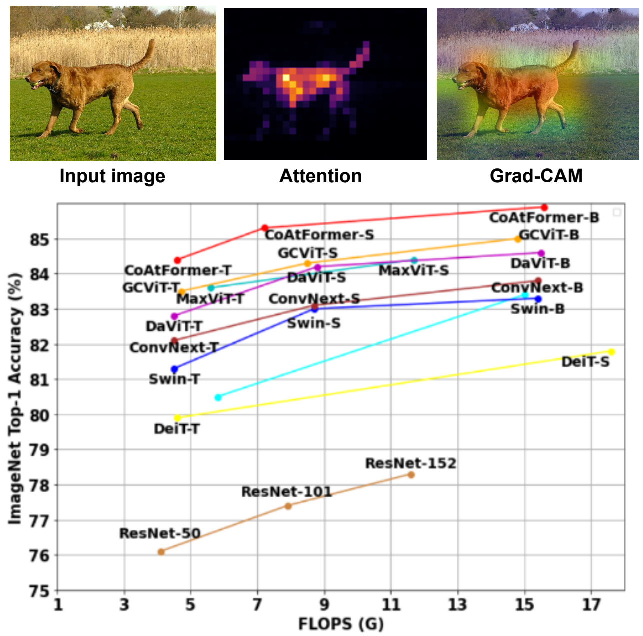



Figure 1: CoAtFormer attains a new state-of-the-art with respect to ImageNet Top-1 accuracy vs computational cost trade-off. For fair comparison, models that are trained and evaluated with input image size of 224 × 224 on ImageNet-1K dataset and without extra data.

demonstrate comparable performance to CNNs. While ViTs present strong capabilities to model the long-range dependencies, their computational complexity grows quadratically with the image size, limiting its application to high resolution images [Yang *et al.*, 2021] in which modeling multi-scale global context is essential for accurate representation modeling.

To reduce the computational cost, some works [Liu *et al.*, 2021; Dong *et al.*, 2021; Tu *et al.*, 2022; Ding *et al.*, 2022], most notably Swin Transformer [Liu *et al.*, 2021], have limited attention region in a spatial local window. However, the limited receptive field of local attention challenges the capability of global self-attention to model global contextual information. Subsequent works such as MaxViT [Tu *et al.*, 2022] attempted to mitigate this issue by developing highly intricate self-attention modules with increased model size. How to efficiently integrate various long-range interactions to balance the model complexity and generalizability under a computation budget still remains challenging.

In this work, we develop a novel Transformer block, called composite attention block, that capably serves as a fundamental architecture component which can model global channel and spatial interactions in a single module. In contrast to vanilla self-attention, composite attention exhibits greater efficiency and flexibility, specifically adapting seamlessly to varying input lengths with linear complexity. Compared to window/local attention, composite attention performs stronger perception capacity by simultaneously capturing global spatial and channel dependencies. Moreover, with only linear complexity, composite attention can serve as a basic stand-alone attention block in any layer of a neural network, even in high-resolution stages.

It is worthwhile to note that our proposed composite attention contains two parallel branches to capture various global context. In composite attention block, one branch exploits a dynamic channels module to model global channel dependencies, while the other branch employs an efficient spatial attention module to capture long-range spatial interactions. Concretely, We replace the self-attention, as originally introduced by Vaswani et al. [Dosovitskiy *et al.*, 2020], with a *efficient spatial attention*. Instead of capturing the pairwise interactions between keys, queries, we model the interactions between a learnable global token embedding and queries only, computing the global context-aware spatial attention weights with linear complexity. Additionally, we use a simple graph neural network as the core component of *dynamic channel attention* to encourage the communication across channels.

Based on the composite attention mechanism, we further propose a simple but effective vision Transformer architecture named "CoAtFormer" by hierarchically stacking repeated blocks composed of composite attention and convolutions. This architecture exhibits significantly stronger modeling power while limiting computation cost. As a general-purpose vision backbone, the CoAtFormer demonstrates state-of-the-art (SOTA) performance for a broad range of visual tasks including image classification, object detection and segmentation. Specifically, for image classification using ImageNet-1K dataset, CoAtFormer with 24M, 37M, 73M parameters achieve new SOTA performance of **84.4%**, **85.3%**, **85.9%** Top-1 accuracy and without using any extra data. As Figure 1 shows, CoAtFormer consistently outperforms both GCViT [Hatamizadeh *et al.*, 2023], MaxViT [Tu *et al.*, 2022] and ConvNeXt [Liu *et al.*, 2022] models by a significant margin. Furthermore, for object detection and instance segmentation using MS COCO dataset, our model achieves a box mAP of **54.9** for object and a mask mAP of **47.5**, surpassing recent state-of-the-art MaxViT counterpart by **+1.5** and **+1.8** respectively.

The main contributions of our work are summarized as follows:

- We propose a strong vision backbone, **CoAtFormer**, that can capture various global interactions throughout every stage of the network.
- We develop a simple but effective composite attention block composed of dynamic channel and efficient spatial attention, enjoining global context in linear complexity.
- Large amounts of performance analysis shows that CoAtFormer outperforms previous SOTA transformers on ImageNet dataset, with a significantly lesser number of parameters. In addition, CoAtFormer also has superior performance on other downstream tasks.

## 2 Related work

**CNNs.** Since AlexNet [Krizhevsky *et al.*, 2012], convolutional neural networks (CNNs) have shown remarkable success in various vision applications [Chen *et al.*, 2017; Tan and Le, 2019; Howard *et al.*, 2017; Sandler *et al.*, 2018; Simonyan and Zisserman, 2014; He *et al.*, 2015]. VGGNet [Simonyan and Zisserman, 2014] and InceptionNets [Szegedy *et al.*, 2014; Szegedy *et al.*, 2015] show that a deep neural network consisted of convolutional layers and pooling layers can attain adequate performance in image recognition. ResNet [He *et al.*, 2015] show stronger generalization ability by introducing skip connections every two layers to the base architecture. ConvNeXt [Liu *et al.*, 2022] has re-introduced core designs of vision transformers and demonstrate a pure CNN can achieve performance comparable to vision transformers on various vision tasks.

**Vision transformers.** Since Transformers [Vaswani *et al.*, 2017; Devlin *et al.*, 2019] achieve tremendous successes in wide natural language processing (NLP) tasks, many efforts [Dosovitskiy *et al.*, 2020; Liu *et al.*, 2021; Dong *et al.*, 2021; Tu *et al.*, 2022; Ding *et al.*, 2022; Hatamizadeh *et al.*, 2023; Yang *et al.*, 2023] have been devoted to developing stronger Transformer based architectures for various vision applications. The pioneering work ViT [Dosovitskiy *et al.*, 2020] directly applies the transformer encoder architecture to a sequence of image patches. However, ViT requires large datasets such as JFT300M [Sun *et al.*, 2017] for training. DeiT [Touvron *et al.*, 2020] utilizes a new training paradigm to enable training of high-performance ViT architecture with fewer data. PVT [Wang *et al.*, 2021] leverages the pyramid structure to generate multi-scale feature maps for generl pixel-level dense prediction tasks. MaxViT [Tu *et al.*, 2022] introduces multi-axis attention to capture both local and global context. DaViT [Ding *et al.*, 2022] presents the dual attention mechanism, which contains spatial self-attention and

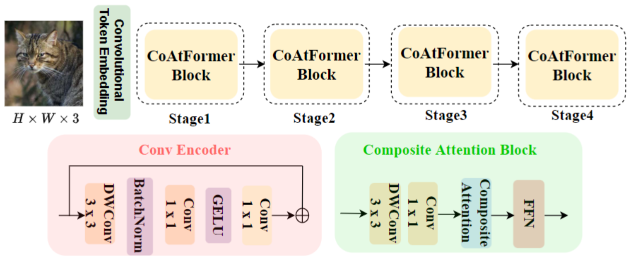



Figure 2: **CoAtFormer architecture.** We follow a typical hierarchical design of CNN practices (e.g., ResNet) but instead build a new type of basic building block that unifies Conv and composite attention blocks. Normalization layers are omitted for simplicity.

channel self-attention. DualViT [Yao *et al.*, 2023] incorporates a critical semantic pathway to obtain global semantics with reduced order of complexity. SMT [Lin *et al.*, 2023] proposes Multi-Head Mixed Convolution (MHMC) module and Scale-Aware Aggregation (SAA) module to enhance the convolutional modulation.

**Channel-wise Attention.** Channel attention mechanisms have been widely adopted in CNNs [Hu *et al.*, 2017; Woo *et al.*, 2018; Qin *et al.*, 2020]. One of the most successful methods is SENet [Hu *et al.*, 2017], which learns channel attention. It first squeezes the feature maps with global average pooling and captures the cross-channel relationships using two fully connected layers. FcaNet [Qin *et al.*, 2020] learns many valuable frequency components by compressing channels using the discrete cosine transform (DCT). Some vision transformers architectures apply channel-wise attention to reduce the computational costs. XCiT [El-Nouby *et al.*, 2021] proposes cross-covariance attention (XCA) to compute channel attention maps. DaViT [Ding *et al.*, 2022] introduces channel group attention (CGA) to perform image-level interactions within each group.

Our work presents dynamic channel attention to capture global channel interactions in Transformers. We demonstrate its power when combined with efficient spatial attention, forming our composite attention mechanism.

# 3 Method

## 3.1 Overall Architecture

We propose Composite Attention Vision Transformers (CoAtFormer), a general, efficient, yet effective Transformer backbone capturing both local and global denpendices. The overall architecture of CoAtFormer is illustrated in Figure 2. For an input image with size of $H \times W \times 3$, we leverage the Convolutional Token Embedding layer (two $3\times3$ convolution layer with a stride 2) to obtain $\frac{H}{4}\times\frac{W}{4}\times C_1$ feature maps. Following the design in modern CNNs (e.g., ResNet [He *et al.*, 2015]), the whole network has four stages to generate feature maps of different scales which are important for dense prediction tasks. To produce the hierarchical representation, A downsampling layer ($3 \times 3$ convolution, stride 2) is used between two consecutive stages to reduce the number of tokens and increase the channel dimension. In each stage, several CoAtFormer blocks are stacked sequentially for feature transformation while maintaining the number of tokens. The CoAtFormer block is able to capture local-global representations.

## 3.2 CoAtFormer Block

The proposed CoAtFormer block contains a *conv encoder* and a *composite attention block*, as illustrated in Figure 3. We will detail these parts in the following.

**Conv Encoder.** The baseline MaxViT [Tu *et al.*, 2022] employs MBConv block [Sandler *et al.*, 2018] as a local token mixer. Although MBConv block has been widely used in efficient models [Tu *et al.*, 2022; Yang *et al.*, 2022a], replacing them with "modernize" convolution block [Liu *et al.*, 2022] does not increase the computational cost. Further, it improves the performance and generalization without increasing the parameters. Using conv encoder before attention provides an additional benefit, as depth-wise convolutions can be considered as conditional position encoding (CPE) [Chu *et al.*, 2021], eliminating the need for explicit positional encoding layers in our model. Specifically, a $3 \times 3$ depth-wise convolution (DWConv) is first applied to capture the local spatial interactions between pixels. Then, the derived features are fed into two point-wise convolutions with GELU activation. Finally, we introduce a residual connection [He *et al.*, 2015] to enable information to flow across the network. Formally, given an input tensor $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$ (H, W, C are its height, width, and channels), the conv encoder is represented as follows:

$$\hat{\mathbf{X}} = \mathrm{BN}(\mathrm{DWConv}_{3\times3}(\mathbf{X})), \tag{1}$$
$$\hat{\mathbf{X}} = \mathrm{Conv}_{1\times1}(\hat{\mathbf{X}}), \tag{2}$$
$$\hat{\mathbf{X}} = \mathrm{GELU}(\hat{\mathbf{X}}), \tag{3}$$
$$\hat{\mathbf{X}} = \mathrm{Conv}_{1\times1}(\hat{\mathbf{X}}) + \mathbf{X}, \tag{4}$$

where BN, GELU, $\mathrm{Conv}_{1\times1}$, and $\mathrm{DWConv}_{3\times3}$ denote Batch Normalization [Ioffe and Szegedy, 2015], Gaussian error Linear Unit [Hendrycks and Gimpel, 2016], $1 \times 1$ point-wise convolution, and $3 \times 3$ depth-wise convolution, respectively.

**Composite Attention Block.** The detailed architecture of the composite attention block is presented in Figure 3. The composite attention block aims to learn enriched local-global features. It begins with local convolutional layers to extract local representations, followed by the composite attention mechanism. As the core element in a composite attention block, the composite attention contains two parallel branches. The two branches share the same input, but focus on relationships of different global context, which can be complementary to each other. In each branch, we first extract global channel or spatial dependencies using Dynamic Channel Attention (DCA) Module or Efficient Spatial Attention (ESA) Module, respectively, then the outputs of two branches can be merged using concatenation. We then describe our dynamic channel attention module and efficient spatial attention module.

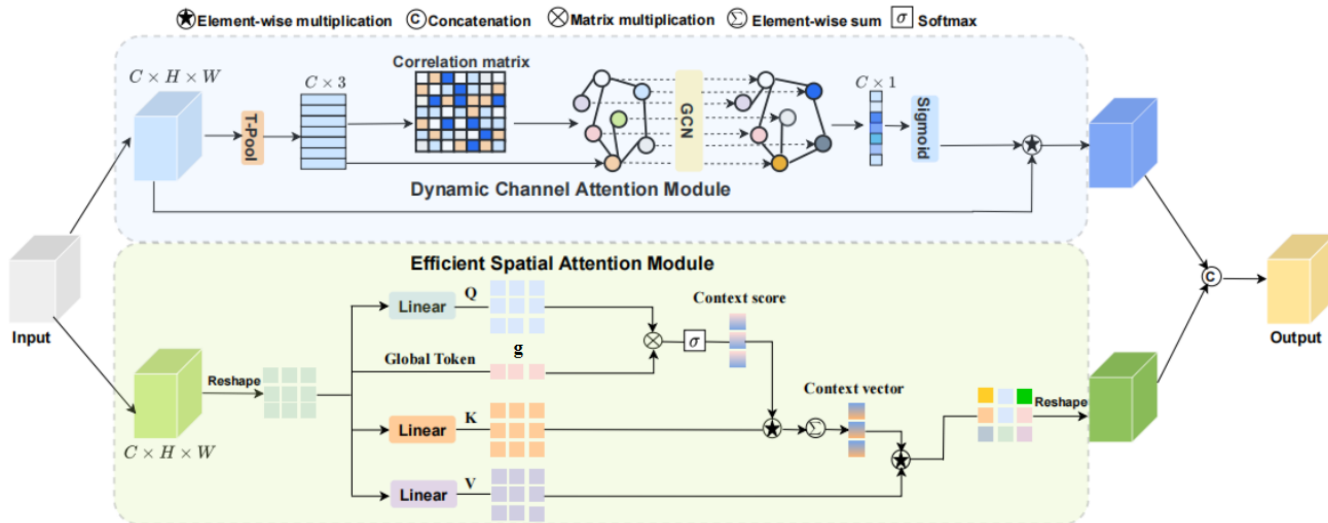



Figure 3: **Composite attention.** An illustration of the composite attention block for computing spatial and channel attention. The **dynamic channel attention module** captures global channel interactions through a simple graph neural network, while the **efficient spatial attention module** is able to compute the global context-aware spatial attention weights with linear complexity.

**Dynamic Channel Attention Module.** In Figure 3, the top branch in composite attention aims to model cross-channel interactions. For an input feature tensor $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, we first squeeze global spatial information into a channel descriptor. This is achieved by using T-Pool layer to reduce the spatial dimension of the input tensor to three by concatenating the average pooled, max pooled and std pooled features across that dimension. Mathematically, the channel responses $\mathbf{T} \in \mathbb{R}^{C\times 3}$ are calculated by:

$$\mu_{\mathrm{c}} = \frac{1}{\mathrm{HW}} \sum_{\mathrm{i}=1}^{\mathrm{H}} \sum_{\mathrm{j}=1}^{\mathrm{W}} \mathbf{x}_{\mathrm{c,i,j}}, \tag{5}$$

$$\sigma_{\mathrm{c}} = \sqrt{\frac{1}{\mathrm{HW}} \sum_{\mathrm{i}=1}^{\mathrm{H}} \sum_{\mathrm{j}=1}^{\mathrm{W}} (\mathbf{x}_{\mathrm{c,i,j}} - \mu_{\mathrm{c}})^2}, \tag{6}$$

$$\nu_{\mathrm{c}} = \max_{i,j} \mathbf{x}_{c,i,j} \tag{7}$$

$$\mathbf{t}_{\mathrm{c}} = [\mu_c, \sigma_c, \nu_c]\,. \tag{8}$$

These channel features can be viewed as a set of unordered nodes which are denoted as $\mathcal{V} = \{v_1, v_2, ...v_C\}$. We then can build a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{E}$ stands for all the edges. Each edge reflects the relation weight between two nodes. Based on this graph, we employ a double-layer graph convolution network [Kipf and Welling, 2016] to propagate information between nodes, the information diffusion process can be expressed as:

$$\mathbf{F} = \mathrm{h}(\mathbf{A}\mathrm{h}(\mathbf{ATW}_1)\mathbf{W}_2), \tag{9}$$

where $\mathbf{W}_1 \in \mathbb{R}^{3\times d_1}$ and $\mathbf{W}_2 \in \mathbb{R}^{d_1\times d_2}$ are state update matrix to be learned, h denotes a non-linear activation, which is GELU [Hendrycks and Gimpel, 2016] in our work. $\mathbf{A} \in \mathbb{R}^{C\times C}$ is a correlation matrix for propagating information, which contains the relation weight between nodes. In our experiments, A is randomly initialized and trained in an end-to-end manner along with the whole model. After graph-level processing, we aggregate node features $\mathbf{F} \in \mathbb{R}^{C\times d_2}$ by taking average of node feature map on the feature dimension then exploit a sigmoid activation to generate channel-wise weights:

$$\mathbf{S} = \mathrm{sigmoid}(\frac{1}{\mathrm{d}_2} \sum_{\mathrm{i}=1}^{\mathrm{d}_2} \mathbf{F}_{\mathrm{c,i}}). \tag{10}$$

Consequently, the output feature maps of DCA module can be formulated as:

$$\mathbf{Y}_{\mathrm{dca}} = \mathbf{X} * \mathbf{S}, \tag{11}$$

where $*$ denotes element-wise multiplication.

Compared to the recent SOTA channel group attention (CGA) [Ding *et al.*, 2022], our proposed DCA enables a more comprehensive communication through a double-layer graph convolution network by treating each channel as a node in the graph. We will verify the effectiveness of DCA in subsequent sections.

**Efficient Spatial Attention Module.** In Figure 3, the bottom branch of composite attention focuses on global spatial context. For an input feature tensor $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, we reshape it to get the input token embedding $\mathbf{X} \in \mathbb{R}^{n\times d}$, where $n = H \times W$ is the number of patches, $d = C$ is the dimensions of the token embedding. The input token embedding $\mathbf{X}$ is linearly transformed into query $\mathbf{Q}$, key $\mathbf{K}$, and value $\mathbf{V}$ using three matrices $\mathbf{W_q}$, $\mathbf{W_k}$, and $\mathbf{W_v}$, where $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{n\times d}$, $\mathbf{W_q}, \mathbf{W_k}, \mathbf{W_v} \in \mathbb{R}^{d\times d}$. First, we compute the dot products of the query with a learnable global token vector $\mathbf{g} \in \mathbb{R}^{d}$, and then apply a softmax function to produce the global context scores $\mathbf{c}_s \in \mathbb{R}^{n}$ as:

$$\alpha = \mathrm{softmax}(\frac{\mathbf{Q}.\mathbf{g}}{\sqrt{d}}). \quad (12)$$

The global context scores are able to capture the importance of each query element. Next, the global context scores are multiplied by the key and pooled, resulting in a single global context vector as follows:

$$\mathbf{c} = \sum_{i=1}^{n} \alpha_i * \mathbf{K}_i. \quad (13)$$

Then, the global context vector is multiplied element-wise with the value to form the global spatial information and fed to a linear layer with weights $\mathbf{W_o} \in \mathbb{R}^{d\times d}$ to obtain the output tensor $\mathbf{Y}_{esa}$, it can be described as:

$$\mathbf{Y}_{esa} = (\mathbf{V} * \mathbf{c})\mathbf{W_o}. \quad (14)$$

Finally, the output tensor $\mathbf{Y}_{esa}$ is reshaped to the original input feature dimensions ($C \times H \times W$). Our proposed ESA module is comparatively cheap to compute compared to vanilla MHSA [Dosovitskiy *et al.*, 2020] and has linear complexity with the token length.

After obtaining the outputs from these two branches, we concatenate them along the channel dimension and then project the result back to the original dimension:

$$\mathbf{Y}_{merged} = \mathrm{concat}(\mathbf{Y}_{dca}, \mathbf{Y}_{esa})\mathbf{W_m}, \quad (15)$$

where, $\mathbf{W_m} \in \mathbb{R}^{2C\times C}$ is a weight matrix. We employ the FFN of vanilla ViT [Dosovitskiy *et al.*, 2020], which consists of two linear layers and a GELU activation.

With the aforementioned these components, the CoAtFormer block can be formulated as:

$$\begin{aligned} \mathbf{Y}_{\mathrm{l}} &= \text{Conv-Encoder}(\mathbf{X}_{l-1}), && (16)\\ \mathbf{M}_{\mathrm{l}} &= \mathrm{DWConv}_{3\times3}(\mathrm{Conv}_{1\times1}(\mathbf{Y}_l)), && (17)\\ \mathbf{Z}_{\mathrm{l}} &= \text{Composite-Attention}(\mathrm{BN}(\mathbf{M}_{\mathrm{l}})) + \mathbf{M}_{\mathrm{l}}, && (18)\\ \mathbf{X}_{\mathrm{l}} &= \mathrm{FFN}(\mathrm{BN}(\mathbf{Z}_{\mathrm{l}})) + \mathbf{Z}_{\mathrm{l}}, && (19) \end{aligned}$$

where $\mathbf{Y}_l$, $\mathbf{M}_l$ and $\mathbf{Z}_l$ denote the intermediate output features in the l-th block, respectively. BN denotes the batch normalization [Ioffe and Szegedy, 2015].

### 3.3 Architecture Variants

For a fair comparison with the other vision Transformer, we consider four different models with various number of parameters and computational complexity. Specifically, we introduce CoAtFormer-T(iny), CoAtFormer-S(mall), CoAtFormer-B(ase) and CoAtFormer-L(arge) variants, which is corresponded to CoAtFormer-T, CoAtFormer-S, CoAtFormer-B and CoAtFormer-L, respectively. In all these variants, the expansion ratio of each FFN is set as 3. The detail configurations of basic embedding channels C and number of blocks $N_i$ are presented as following:

- CoAtFormer-T:$C = 64, N_i = \{1, 3, 12, 1\}$
- CoAtFormer-S:$C = 80, N_i = \{1, 3, 12, 1\}$
- CoAtFormer-B:$C = 96, N_i = \{1, 6, 18, 1\}$
- CoAtFormer-L:$C = 128, N_i = \{1, 10, 22, 1\}$

| Model | Year | Params | FLOPs | Top-1 Acc (%) |
|---|---|---|---|---|
| DeiT-S/16 [Touvron *et al.*, 2020] | ICML21 | 22.1M | 4.5G | 79.8 |
| PVT-S [Wang *et al.*, 2021] | ICCV21 | 24.5M | 3.8G | 79.8 |
| Swin-T [Liu *et al.*, 2021] | ICCV21 | 28.3M | 4.5G | 81.2 |
| ConvNeXt-T [Liu *et al.*, 2022] | CVPR22 | 28.6M | 4.5G | 82.1 |
| DaViT-T [Ding *et al.*, 2022] | ECCV22 | 28.3M | 4.5G | 82.8 |
| MOAT-0 [Yang *et al.*, 2022a] | ICLR23 | 27.8M | 5.7G | 83.3 |
| GPViT-L2 [Hatamizadeh *et al.*, 2023] | ICLR23 | 23.8M | 15.0G | 83.4 |
| Dual-ViT-S [Yao *et al.*, 2023] | TPAMI23 | 24.6M | 4.8G | 83.4 |
| GCViT-T [Hatamizadeh *et al.*, 2023] | ICML23 | 28.0M | 4.7G | 83.5 |
| MaxViT-T [Tu *et al.*, 2022] | ECCV22 | 31.0M | 5.6G | 83.6 |
| SMT-S [Lin *et al.*, 2023] | ICCV23 | 20.5M | 4.7G | 83.7 |
| **CoAtFormer-T (Ours)** | | 24.3M | 4.5G | **84.4** |
| Swin-S [Liu *et al.*, 2021] | ICCV21 | 50.0M | 8.7G | 83.0 |
| ConvNeXt-S [Liu *et al.*, 2022] | CVPR22 | 50.0M | 8.7G | 83.1 |
| CSwin-S [Dong *et al.*, 2021] | CVPR22 | 35.0M | 6.9G | 83.6 |
| DaViT-S [Ding *et al.*, 2022] | ECCV22 | 49.7M | 8.8G | 84.2 |
| MOAT-1 [Yang *et al.*, 2022a] | ICLR23 | 41.6M | 9.7G | 84.2 |
| CrossrFormer++-B [Wang *et al.*, 2023] | TPAMI23 | 52.0M | 9.5G | 84.2 |
| GPViT-L3 [Hatamizadeh *et al.*, 2023] | ICLR23 | 36.2M | 22.9G | 84.1 |
| GCViT-S [Hatamizadeh *et al.*, 2023] | ICML23 | 51.0M | 8.5G | 84.3 |
| MaxViT-S [Tu *et al.*, 2022] | ECCV22 | 69.0M | 11.7G | 84.4 |
| Dilate-S [Jiao *et al.*, 2023] | TMM23 | 49.0M | 10.0G | 84.4 |
| SMT-B [Lin *et al.*, 2023] | ICCV23 | 32.0M | 7.7G | 84.3 |
| **CoAtFormer-S (Ours)** | | 37.5M | 7.2G | **85.3** |
| Swin-B [Liu *et al.*, 2021] | ICCV21 | 88.0M | 15.4G | 83.3 |
| ConvNeXt-B [Liu *et al.*, 2022] | CVPR22 | 89.0M | 15.4G | 83.8 |
| HiViT-B [Zhang *et al.*, 2023] | ICLR23 | 66.4M | 15.9G | 83.8 |
| CSwin-B [Dong *et al.*, 2021] | CVPR22 | 78.0M | 15.0G | 84.2 |
| GPViT-L4 [Hatamizadeh *et al.*, 2023] | ICLR23 | 75.4M | 48.2G | 84.3 |
| DaViT-B [Ding *et al.*, 2022] | ECCV22 | 87.9M | 15.5G | 84.6 |
| CrossFormer++-L [Wang *et al.*, 2023] | TPAMI23 | 92.0M | 16.6G | 84.7 |
| MOAT-2 [Yang *et al.*, 2022a] | ICLR23 | 73.4M | 17.2G | 84.7 |
| MaxViT-B [Tu *et al.*, 2022] | ECCV22 | 120.0M | 74.2G | 84.9 |
| GCViT-B [Hatamizadeh *et al.*, 2023] | ICML23 | 90.0M | 14.8G | 85.0 |
| **CoAtFormer-B (Ours)** | | 73.4M | 15.4G | **85.9** |
| Pre-trained on ImageNet-22k | | | | |
| Swin-L [Liu *et al.*, 2021] | ICCV21 | 196.5M | 34.5G | 86.3 |
| Swin-L [Liu *et al.*, 2021]† | ICCV21 | 196.5M | 103.9G | 87.3 |
| ConvNeXt-L [Liu *et al.*, 2022] | CVPR22 | 198.0M | 34.4G | 86.6 |
| ConvNeXt-L [Liu *et al.*, 2022]† | CVPR22 | 198.0M | 101.0G | 87.5 |
| CSwin-L [Dong *et al.*, 2021] | CVPR22 | 173.0M | 31.5G | 86.5 |
| CSwin-L [Dong *et al.*, 2021]† | CVPR22 | 173.0M | 96.8G | 87.5 |
| MOAT-3 [Yang *et al.*, 2022a] | ICLR23 | 190.0M | 44.9G | 86.8 |
| MOAT-3 [Yang *et al.*, 2022a]† | ICLR23 | 190.0M | 141.2G | 88.2 |
| GCViT-L [Hatamizadeh *et al.*, 2023] | ICML23 | 201.0M | 32.6G | 86.6 |
| DaViT-L [Ding *et al.*, 2022]† | ECCV22 | 196.8M | 103.0G | 87.5 |
| MaxViT-L [Tu *et al.*, 2022]† | ECCV22 | 212.0M | 128.7G | 88.3 |
| **CoAtFormer-L (Ours)** | | 157.6M | 36.2G | **87.6** |
| **CoAtFormer-L (Ours)†** | | 157.6M | 105.8G | **88.7** |

Table 1: Comparison of image classification on ImageNet-1K for different models. All models are trained and evaluated with 224×224 resolution on ImageNet-1K by default, unless otherwise noted. † denotes the model is evaluated with resolution of 384 × 384.

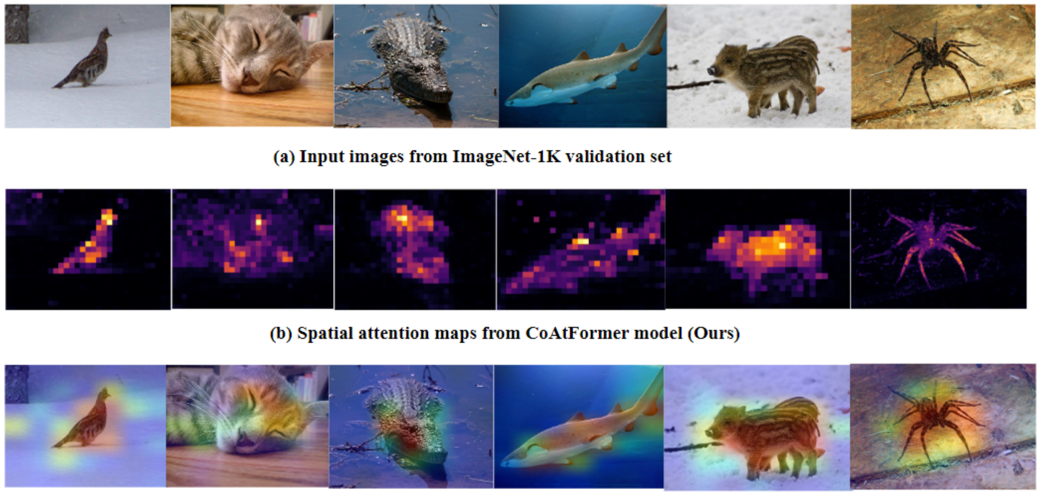
## 4 Experiments

To validate the efficacy of CoAtFormer as a general vision backbone, we conduct experiments on ImageNet-1K [Deng *et al.*, 2009] classification, COCO [Lin *et al.*, 2014] object detection and instance segmentation, and ADE20K [Zhou *et al.*, 2017] semantic segmentation. We also perform comprehensive ablation studies to evaluate the effectiveness of each component of CoAtFormer.

### 4.1 Image Classification

**Setup.** For fair comparison, we follow the same training strategies as previous works [Touvron *et al.*, 2020; Liu *et al.*, 2021]. Specifically, we train all our models for 300 epochs with the input size of 224×224. We employ the AdamW optimizer with weight decay of 0.05. The default batch size and initial learning rate are set to 1024 and 0.001. Additionally, we explore the effectiveness of our models when pretrained on ImageNet-22K.

**Results.** In Table 1, we compare our proposed CoAtFormer with current state-of-the-art models. It shows that



(a) Input images from ImageNet-1K validation set

(b) Spatial attention maps from CoAtFormer model (Ours)

(c) GradCAM maps

Figure 4: **Visualization**: (a) input images (b) efficient spatial attention maps from CoAtFormer-T model (c) corresponding Grad-CAM attention maps.

our CoAtFormer attains new state-of-the-art and consistently outperforms other models by large margins. Specifically, CoAtFormer-T achieves 84.4% Top-1 accuracy with only 4.6G FLOPs, surpassing Swin-T, DaViT-T and MaxViT-T by 3.2%, 1.6% and 0.8% respectively. In addition, when compared to small-sized and base-sized models, our CoAtFormer also achieves the best performance. Notably, CoAtFormer-S achieves a top-1 accuracy of 85.3% with only 37.5M parameters and 7.2GFLOPs of computation, outperforming many counterpart-Base models such as Swin-B [Liu *et al.*, 2021], ConvNeXt-B [Liu *et al.*, 2022], and MaxViT-B [Tu *et al.*, 2022], which have over 80M parameters and 15GFLOPs of computation. For example, our CoAtFormer-S attains 2.0% and 0.7% higher accuracy than Swin-B and DaViT-B, respectively, using near half computations.

We further present the ImageNet-22K pre-training results in Table 1. CoAtFormer-L achieves an 88.7% top-1 accuracy, surpassing MOAT-3 [Yang *et al.*, 2022a] by 0.5% while utilizing fewer parameters (157.5M vs. 190M) and lower FLOPs (105.8G vs. 141.2G). This highly highlights the remarkable scalability capabilities of CoAtFormer.

## 4.2 Object Detection and Instance Segmentation

**Setup.** We evaluate our models on object detection with COCO 2017 [Lin *et al.*, 2014]. All the backbones are first pretrained using ImageNet-1K. The pretrained models are then plugged into two detectors, Mask R-CNN [He *et al.*, 2017b] and Cascade Mask R-CNN [Cai and Vasconcelos, 2017] . We follow the finetuning strategy used in ConvNeXt [Liu *et al.*, 2022] on the COCO training set.

**Results.** Table 2 reports the results of our models on MS COCO dataset. Using a Mask-RCNN detector, our CoAtFormer-T (49.1/44.8) backbone outperforms counterparts with ConvNeXt-T (46.2/41.7) by **+2.9** and **+3.1** and DaViT-T (47.4/42.9) by **+1.7** and **+1.9** in terms of box AP and mask AP, respectively. Using a Cascade Mask-RCNN detector, we also observe substantial gains across all model configurations. Furthermore, we observe a almost saturated mAP in Swin Transformer [Liu *et al.*, 2021] and GCViT [Hatamizadeh *et al.*, 2023] from small to base model, while the mAP of our model consistently improves with larger model size, demonstrating enhanced scalability.

| Backbone | Param (M) | FLOPs (G) | $AP^{box}$ | $AP^{box}_{50}$ | $AP^{mask}$ | $AP^{mask}_{50}$ |
|---|---|---|---|---|---|---|
| Mask-RCNN 3× schedule | | | | | | |
| PVT-S [Wang *et al.*, 2021] | 44 | 245 | 43.0 | 65.3 | 39.9 | 62.5 |
| Swin-T [Liu *et al.*, 2021] | 48 | 267 | 46.0 | 68.1 | 41.6 | 65.1 |
| ConvNeXt-T [Liu *et al.*, 2022] | 48 | 262 | 46.2 | 67.9 | 41.7 | 65.0 |
| DaViT-T [Ding *et al.*, 2022] | 48 | 263 | 47.4 | 69.5 | 42.9 | 66.8 |
| GCViT-T [Hatamizadeh *et al.*, 2023] | 48 | 291 | 47.9 | 70.1 | 43.2 | 67.0 |
| **CoAtFormer-T** | 44 | 263 | **49.1** | **70.6** | **44.8** | **67.6** |
| Cascade Mask-RCNN 3× schedule | | | | | | |
| DeiT-S/16 [Touvron *et al.*, 2020] | 80 | 889 | 48.0 | 67.2 | 41.4 | 64.2 |
| ResNet-50 [He *et al.*, 2015] | 82 | 739 | 46.3 | 64.3 | 40.1 | 61.7 |
| Swin-T [Liu *et al.*, 2021] | 86 | 745 | 50.4 | 69.2 | 43.7 | 66.6 |
| ConvNeXt-T [Liu *et al.*, 2022] | 86 | 741 | 50.4 | 69.1 | 43.7 | 66.5 |
| GCViT-T [Hatamizadeh *et al.*, 2023] | 85 | 770 | 51.6 | 70.4 | 44.6 | 67.8 |
| **CoAtFormer-T** | 81 | 742 | **52.8** | **71.4** | **46.3** | **69.2** |
| Swin-S [Liu *et al.*, 2021] | 107 | 838 | 51.9 | 70.7 | 45.0 | 68.2 |
| ConvNeXt-S [Liu *et al.*, 2022] | 108 | 827 | 51.9 | 70.8 | 45.0 | 68.4 |
| GCViT-S [Hatamizadeh *et al.*, 2023] | 108 | 866 | 52.4 | 71.0 | 45.4 | 68.5 |
| **CoAtFormer-S** | 96 | 825 | **53.5** | **72.8** | **47.1** | **70.7** |
| Swin-B [Liu *et al.*, 2021] | 145 | 982 | 51.9 | 70.5 | 45.0 | 68.1 |
| ConvNeXt-B [Liu *et al.*, 2022] | 146 | 964 | 52.7 | 71.3 | 45.6 | 68.9 |
| GCViT-B [Hatamizadeh *et al.*, 2023] | 146 | 1018 | 52.9 | 71.7 | 45.8 | 69.2 |
| MaxViT-B [Tu *et al.*, 2022] | 157 | 856 | 53.4 | 72.9 | 45.7 | 70.3 |
| **CoAtFormer-B** | 138 | 960 | **54.9** | **74.1** | **47.5** | **71.2** |

Table 2: Object detection and instance segmentation performance on the COCO val2017 with Mask R-CNN and Cascade Mask R-CNN. All models are pretrained on ImageNet-1K.

## 4.3 Semantic Segmentation on ADE20k

**Setup.** We further benchmark our method for semantic segmentation using the ADE20K dataset. We use UperNet [Xiao *et al.*, 2018] as the segmentation method and our CoAtFormer as the backbone. We follow the same training recipe proposed

| Backbone | Param (M) | FLOPs (G) | mIoU |
|---|---|---|---|
| DeiT-Small/16 [Touvron *et al.*, 2020] | 52 | 1099 | 44.0 |
| Swin-T [Liu *et al.*, 2021] | 60 | 945 | 44.5 |
| ResNet-101 [He *et al.*, 2015] | 86 | 1029 | 44.9 |
| ConvNeXt-T [Liu *et al.*, 2022] | 60 | 939 | 46.0 |
| DaViT-T [Ding *et al.*, 2022] | 60 | 940 | 46.3 |
| FocalNet-T [Yang *et al.*, 2022b] | 61 | 944 | 46.5 |
| GCViT-T [Hatamizadeh *et al.*, 2023] | 58 | 947 | 47.0 |
| **CoAtFormer-T** | 56 | 940 | **48.8** |
| Swin-S [Liu *et al.*, 2021] | 81 | 1038 | 47.6 |
| GCViT-S [Hatamizadeh *et al.*, 2023] | 84 | 1163 | 48.3 |
| ConvNeXt-S [Liu *et al.*, 2022] | 84 | 1027 | 48.7 |
| DaViT-S [Ding *et al.*, 2022] | 81 | 1030 | 48.8 |
| FocalNet-S [Yang *et al.*, 2022b] | 83 | 1035 | 49.3 |
| Daliate-B [Jiao *et al.*, 2023] | 79 | 1046 | 50.8 |
| **CoAtFormer-S** | 70 | 1012 | **51.3** |
| Swin-B [Liu *et al.*, 2021] | 121 | 1188 | 48.1 |
| ConvNeXt-B [Liu *et al.*, 2022] | 122 | 1170 | 49.1 |
| GCViT-B [Hatamizadeh *et al.*, 2023] | 125 | 1348 | 49.2 |
| DaViT-B [Ding *et al.*, 2022] | 121 | 1175 | 49.4 |
| FocalNet-B [Yang *et al.*, 2022b] | 124 | 1180 | 50.2 |
| **CoAtFormer-B** | 106 | 1173 | **52.1** |

Table 3: Comparison with SoTA methods for semantic segmentation on ADE20K val set. Single-scale inference is used. FLOPs are measured by $512 \times 2048$.

| Model | Param (M) | FLOPs (G) | Top-1 (%) |
|---|---|---|---|
| DCA $\rightarrow$ ESA | 23.1 | 4.4 | 84.0 |
| ESA $\rightarrow$ DCA | 23.1 | 4.4 | 84.1 |
| Parallel branch (ours) | 24.3 | 4.5 | **84.4** |

Table 4: Quantitative comparisons of different composite attention layouts on ImageNet-1K.

by [Liu *et al.*, 2021]. Specifically, we train UperNet for 160k iterations with an input size of $512 \times 512$. We use the AdamW optimizer with a weight decay of 0.01 and set the batch size to 16.

**Results.** In Table 3, the results show that our CoAtFormer outperforms DaViT, FocalNet, and GCViT significantly under all configurations. Concretely, CoAtFormer-T (48.8), CoAtFormer-S (51.3) and CoAtFormer-B (52.1) backbones improve **1.8**, **2.5** and **1.9** mIoU gains over counterpart models with GCViT [Hatamizadeh *et al.*, 2023], respectively.

### 4.4 Ablation Study

**Composite attention layout.** We perform experiments on the arrangement of our composite attention mechanism. Three options with comparable computations are explored: (i) dynalic channel attention (DCA) first; (ii) efficient spatial attention (ESA) first; and (iii) parallel arrangement of both types of attention. The results are presented in Table 4. It can be observed that the three strategies yield similar performance, with a slight advantage for the 'parallel arrangement of both types of attention' approach.

**Component analysis.** Table 5 demonstrates the significance of composite attention block and conv encoder in our

| | Model | Param | FLOPs | Top-1 (%) |
|---|---|---|---|---|
| Baseline | CoAtFormer-T | 24.3M | 4.5G | **84.4** |
| Different Components | w/o Conv Encoder | 22.0M | 4.1G | 83.4 |
| | w/o Composite Attention | 17.6M | 3.3G | 81.5 |
| Composite Attention Components | w/o Dynamic Channel Attention | 22.0M | 4.2G | 83.1 |
| | w/o Efficient Spatial Attention | 17.6M | 3.3G | 82.9 |

Table 5: Ablation on different components of CoAtFormer and composite attention block. The results show the benefits of composite attention block in our design.

| Fusion method | Param (M) | FLOPs (G) | Top-1 (%) |
|---|---|---|---|
| Concat & project | 24.3 | 4.5 | **84.4** |
| Sum | 19.5 | 3.6 | 83.5 |
| Weighted sum | 19.6 | 3.6 | 83.6 |

Table 6: Comparison of different fusion methods in composite attention mechanism.

proposed architecture. Substituting composite attention block with convolution encoder leads to a decrease in accuracy by 2.9%, highlighting the effectiveness of composite attention block in our design. Moreover, removing conv encoder in all four stages of the network results in an additional accuracy reduction of 1.0%. In addition, We also measure the contributions of composite attention components (e.g., dynamic channel attention (DCA) and efficient spatial attention (ESA) ) in Table 5. Removing DCA module and ESA module decrease the accuracy by 1.3 and 1.5, respectively.

**Different fusion method.** We try two general operations as the alternatives to the fusion method in composite attention. One is element-wise addition, the other is weighted sum. The results are shown in Table 6. It is clear that concatenation and an project layer as the fusion module achieves the best model size and accuracy trade-off among the different fusion methods we evaluated.

## 5 Interpretability

In Figure 4, we can find that learned spatial attention distributions align the region of image semantics, and hence demonstrate the effectiveness of efficient spatial attention. Additionally, corresponding Grad-CAM maps present accurate object localization with most intricate details.

## 6 Conclusion

In this work, we propose a novel architecture named CoAtFormer, which can efficiently capture global channel and spatial contexts by utilizing dynamic channel attention and efficient spatial attention. The proposed CoAtFormer architectures take advantages of both CNNs and transformers to capture local and global information, improving the representation ability of the model. Extensive experiments on on ImageNet and other downstream vision applications demonstrate the effectiveness and superiority of the proposed architecture.

## References


[Cai and Vasconcelos, 2017] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[Chen *et al.*, 2017] Yunpeng Chen, Jianan Li, Huaxin Xiao, Xiaojie Jin, Shuicheng Yan, and Jiashi Feng. Dual path networks. In *Neural Information Processing Systems*, 2017.

[Chu *et al.*, 2021] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, and Chunhua Shen. Conditional positional encodings for vision transformers. In *International Conference on Learning Representations*, 2021.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019.

[Ding *et al.*, 2022] Mingyu Ding, Bin Xiao, Noel C. F. Codella, Ping Luo, Jingdong Wang, and Lu Yuan. Davit: Dual attention vision transformers. *ArXiv*, abs/2204.03645, 2022.

[Dong *et al.*, 2021] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. Cswin transformer: A general vision transformer backbone with cross-shaped windows. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[Dosovitskiy *et al.*, 2020] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020.

[El-Nouby *et al.*, 2021] Alaaeldin El-Nouby, Hugo Touvron, Mathilde Caron, Piotr Bojanowski, Matthijs Douze, Armand Joulin, Ivan Laptev, Natalia Neverova, Gabriel Synnaeve, Jakob Verbeek, and Hervé Jégou. Xcit: Cross-covariance image transformers. In *Neural Information Processing Systems*, 2021.

[Hatamizadeh *et al.*, 2023] Ali Hatamizadeh, Hongxu Yin, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. Global context vision transformers. In *International Conference on Machine Learning*. PMLR, 2023.

[He *et al.*, 2015] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[He *et al.*, 2017a] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42, 2017.

[He *et al.*, 2017b] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. 2017.

[Hendrycks and Gimpel, 2016] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv: Learning*, 2016.

[Howard *et al.*, 2017] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv*, abs/1704.04861, 2017.

[Hu *et al.*, 2017] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, abs/1502.03167, 2015.

[Jiao *et al.*, 2023] Jiayu Jiao, Yu-Ming Tang, Kun-Yu Lin, Yipeng Gao, Jinhua Ma, Yaowei Wang, and Wei-Shi Zheng. Dilateformer: Multi-scale dilated transformer for visual recognition. *IEEE Transaction on Multimedia*, 2023.

[Kipf and Welling, 2016] Thomas Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ArXiv*, 2016.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60, 2012.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

[Lin *et al.*, 2023] Wei-Shiang Lin, Ziheng Wu, Jiayu Chen, Jun Huang, and Lianwen Jin. Scale-aware modulation meet transformer. *ArXiv*, abs/2307.08579, 2023.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[Liu *et al.*, 2022] Zhuang Liu, Hanzi Mao, Chaozheng Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[Qin *et al.*, 2020] Zequn Qin, Pengyi Zhang, Fei Wu, and Xi Li. Fcanet: Frequency channel attention networks. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2020.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 2015.

[Sandler *et al.*, 2018] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[Sun *et al.*, 2017] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Kumar Gupta. Revisiting unreasonable effectiveness of data in deep learning era. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[Szegedy *et al.*, 2014] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, D. Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[Szegedy *et al.*, 2015] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[Tan and Le, 2019] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *ArXiv*, abs/1905.11946, 2019.

[Touvron *et al.*, 2020] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herv'e J'egou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, 2020.

[Tu *et al.*, 2022] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Conrad Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. In *European Conference on Computer Vision*, 2022.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Neural Information Processing Systems*, 2017.

[Wang *et al.*, 2017] X. Wang, Ross B. Girshick, Abhinav Kumar Gupta, and Kaiming He. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[Wang *et al.*, 2021] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[Wang *et al.*, 2023] Wenxiao Wang, Wei Chen, Qibo Qiu, Long Chen, Boxi Wu, Binbin Lin, Xiaofei He, and Wei Liu. Crossformer++: A versatile vision transformer hinging on cross-scale attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence, TPAMI*, 2023.

[Woo *et al.*, 2018] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In-So Kweon. Cbam: Convolutional block attention module. In *European Conference on Computer Vision*, 2018.

[Xiao *et al.*, 2018] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. *ArXiv*, abs/1807.10221, 2018.

[Yang *et al.*, 2021] Huanrui Yang, Hongxu Yin, Pavlo Molchanov, Hai Li, and Jan Kautz. Nvit: Vision transformer compression and parameter redistribution. *ArXiv*, abs/2110.04869, 2021.

[Yang *et al.*, 2022a] Chenglin Yang, Siyuan Qiao, Qihang Yu, Xiaoding Yuan, Yukun Zhu, Alan Loddon Yuille, Hartwig Adam, and Liang-Chieh Chen. Moat: Alternating mobile convolution and attention brings strong vision models. *ArXiv*, abs/2210.01820, 2022.

[Yang *et al.*, 2022b] Jianwei Yang, Chunyuan Li, Xiyang Dai, and Jianfeng Gao. Focal modulation networks, 2022.

[Yang *et al.*, 2023] Chenhongyi Yang, Jiarui Xu, Shalini De Mello, Elliot J. Crowley, and Xiaolong Wang. GPViT: A High Resolution Non-Hierarchical Vision Transformer with Group Propagation. 2023.

[Yao *et al.*, 2023] Ting Yao, Yehao Li, Yingwei Pan, Yu Wang, Xiao-Ping Zhang, and Tao Mei. Dual vision transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10870–10882, 2023.

[Zhang *et al.*, 2023] Xiaosong Zhang, Yunjie Tian, Lingxi Xie, Wei Huang, Qi Dai, Qixiang Ye, and Qi Tian. Hivit: A simpler and more efficient design of hierarchical vision transformer. In *International Conference on Learning Representations*, 2023.

[Zheng *et al.*, 2020] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[Zhou *et al.*, 2017] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[Zhu *et al.*, 2020] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *ArXiv*, abs/2010.04159, 2020.