Scale-Aware Modulation Meet Transformer

Weifeng Lin[1,2], Ziheng Wu[2], Jiayu Chen[2], Jun Huang [2], Lianwen Jin[1*]

[1] South China University of Technology, [2] Platform of AI (PAI), Alibaba Group

eelinweifeng@mail.scut.edu.cn   eelwjin@scut.edu.cn   {ziheng.wzh, yunji.cjy, huangjun.hj}@alibaba-inc.com

Abstract

*This paper presents a new vision Transformer, Scale-Aware Modulation Transformer (SMT), that can handle various downstream tasks efficiently by combining the convolutional network and vision Transformer. The proposed Scale-Aware Modulation (SAM) in the SMT includes two primary novel designs. Firstly, we introduce the Multi-Head Mixed Convolution (MHMC) module, which can capture multi-scale features and expand the receptive field. Secondly, we propose the Scale-Aware Aggregation (SAA) module, which is lightweight but effective, enabling information fusion across different heads. By leveraging these two modules, convolutional modulation is further enhanced. Furthermore, in contrast to prior works that utilized modulations throughout all stages to build an attention-free network, we propose an Evolutionary Hybrid Network (EHN), which can effectively simulate the shift from capturing local to global dependencies as the network becomes deeper, resulting in superior performance. Extensive experiments demonstrate that SMT significantly outperforms existing state-of-the-art models across a wide range of visual tasks. Specifically, SMT with **11.5M / 2.4GFLOPs** and **32M / 7.7GFLOPs** can achieve **82.2%** and **84.3%** top-1 accuracy on ImageNet-1K, respectively. After pretrained on ImageNet-22K in 224² resolution, it attains **87.1%** and **88.1%** top-1 accuracy when finetuned with resolution 224² and 384², respectively. For object detection with Mask R-CNN, the SMT base trained with 1× and 3× schedule outperforms the Swin Transformer counterpart by **4.2** and **1.3** mAP on COCO, respectively. For semantic segmentation with UPerNet, the SMT base test at single- and multi-scale surpasses Swin by **2.0** and **1.1** mIoU respectively on the ADE20K. Our code is available at* https://github.com/AFeng-x/SMT.

1. Introduction

Since the groundbreaking work on Vision Transformers (ViT) [9], Transformers have gained significant atten-
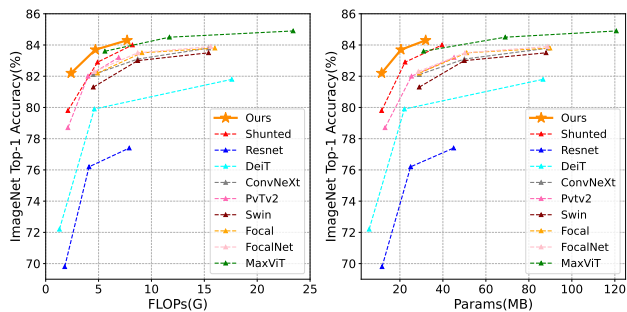

Figure 1: Top-1 accuracy on ImageNet-1K of recent SOTA models. Our proposed SMT outperforms all the baselines.

tion from both industry and academia, achieving remarkable success in various computer vision tasks, such as image classification [8], object detection [27, 10], and semantic segmentation [68, 6]. Unlike convolutional networks, which only allow for interactions within a local region using a shared kernel, ViT divides the input image into a sequence of patches and updates token features via self-attention (SA), enabling global interactions. However, self-attention still faces challenges in downstream tasks due to the quadratic complexity in the number of visual tokens, particularly for high-resolution inputs.

To address these challenges, several efficient spatial attention techniques have been proposed. For example, Swin Transformer [29] employs window attention to limit the number of tokens and establish cross-window connections via shifting. PVT [52, 53] and Focal [61] reduce the cost of self-attention by combining token merging with spatial reduction. Shunted [38] effectively models objects at multiple scales simultaneously while performing spatial reduction. Other techniques such as dynamic token selection [34, 36, 62] have also proven to be effective improvements.

Rather than directly improving self-attention, several works [7, 24, 33, 23] have investigated hybrid CNN-Transformer architectures that combine efficient convolutional blocks with powerful Transformer blocks. We observed that most hybrid networks replace shallow Trans-

---

*Corresponding authors.

former blocks with convolution blocks to reduce the high computational cost of self-attention in the early stages. However, these simplistic stacking strategies hinder them from achieving a better balance between accuracy and latency. Therefore, one of the objectives of this paper is to present a new perspective on the integration of Transformer and convolution blocks.

Based on the research conducted in [9, 3], which performed a quantitative analysis of different depths of self-attention blocks and discovered that shallow blocks tend to capture short-range dependencies while deeper ones capture long-range dependencies, we propose that substituting convolution blocks for Transformer blocks in shallow networks offers a promising strategy for two primary reasons: (1) self-attention induces significant computational costs in shallow networks due to high-resolution input, and (2) convolution blocks, which inherently possess a capacity for local modeling, are more proficient at capturing short-range dependencies than SA blocks in shallow networks. However, we observed that simply applying the convolution directly to the feature map does not lead to the desired performance. Taking inspiration from recent convolutional modulation networks [13, 16, 60], we discovered that convolutional modulation can aggregate surrounding contexts and adaptively self-modulate, giving it a stronger modeling capability than using convolution blocks alone. Therefore, we proposed a novel convolutional modulation, termed Scale-Aware Modulation (SAM), which incorporates two new modules: Multi-Head Mixed Convolution (MHMC) and Scale-Aware Aggregation (SAA). The MHMC module is designed to enhance the receptive field and capture multi-scale features simultaneously. The SAA module is designed to effectively aggregate features across different heads while maintaining a lightweight architecture. Despite these improvements, we find that SAM falls short of the self-attention mechanism in capturing long-range dependencies. To address this, we propose a new hybrid Modulation-Transformer architecture called the Evolutionary Hybrid Network (EHN). Specifically, we incorporate SAM blocks in the top two stages and Transformer blocks in the last two stages, while introducing a new stacking strategy in the penultimate stage. This architecture not only simulates changes in long-range dependencies from shallow to deep layers but also enables each block in each stage to better match its computational characteristics, leading to improved performance on various downstream tasks. Collectively, we refer to our proposed architecture as Scale-Aware Modulation Transformer (SMT).

As shown in Fig. 1, our SMT significantly outperforms other SOTA vision Transformers and convolutional networks on ImageNet-1K [8]. It is worth noting that our SMT achieves top-1 accuracy of 82.2% and 84.3% with the tiny and base model sizes, respectively. Moreover, our SMT consistently outperforms other SOTA models on COCO [27] and ADE20K [68] for object detection, instance segmentation, and semantic segmentation tasks.

Overall, the contributions of this paper are as follows.

- We introduce the Scale-Aware Modulation (SAM) which incorporates a potent Multi-Head Mixed Convolution (MHMC) and an innovative, lightweight Scale-Aware Aggregation (SAA). The SAM facilitates the integration of multi-scale contexts and enables adaptive modulation of tokens to achieve more precise predictions.

- We propose a new evolutionary hybrid network that effectively models the transition from capturing local to global dependencies as the network increases in depth, leading to improved performance and high efficiency.

- We evaluated our proposed Scale-Aware Modulation Transformer (SMT) on several widely used benchmarks, including classification, object detection, and segmentation. The experimental results indicated that SMT consistently outperformed the SOTA Vision Transformers while requiring fewer parameters and incurring lower computational costs.

2. Related Work

2.1. Vision Transformers

The Transformer [50] was initially developed for natural language processing tasks and has since been adapted for computer vision tasks through the introduction of the Vision Transformer (ViT) [9]. Further improvements to ViT have been achieved through knowledge distillation or more intricate data augmentation, as demonstrated by DeiT [48]. However, Transformers do not consider the quadratic complexity of high-resolution images or the 2D structure of images, which are challenges in vision tasks. To address these issues and improve the performance of vision Transformers, various methods have been proposed, including multi-scale architectures [2, 29, 52, 59], lightweight convolution layers [12, 25, 56], and local self-attention mechanisms [29, 5, 61, 65].

2.2. Convolutional Neural Networks

Convolutional neural networks (CNNs) have been the main force behind the revival of deep neural networks in computer vision. Since the introduction of AlexNet [22], VGGNet [40], and ResNet [15], CNNs have rapidly become the standard framework for computer vision tasks. The design principles of CNNs have been advanced by subsequent models such as Inception [43, 44], ResNeXt [58], Res2Net [11] and MixNet [47], which promote the use of building blocks with multiple parallel convolutional paths. Other works such as MobileNet [18] and ShuffleNet [67]
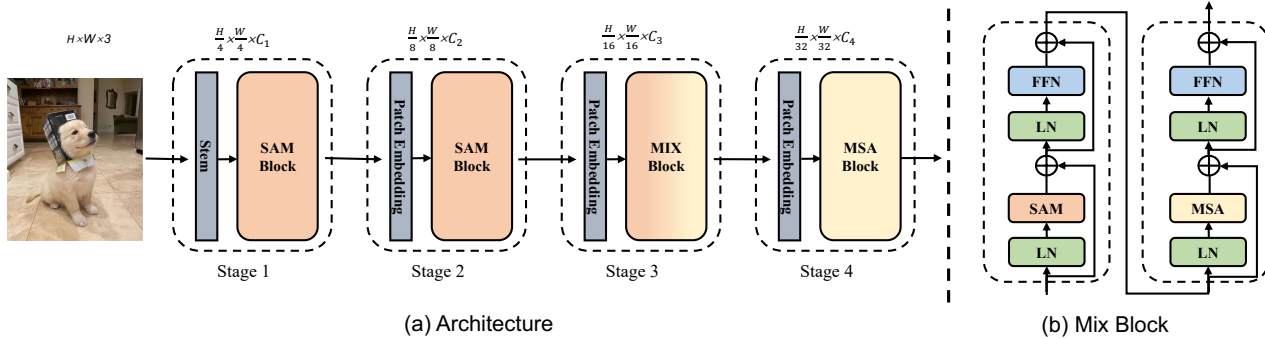
Figure 2: (a) The architecture of the Scale-Aware Modulation Transformer (SMT); (b) Mix Block: a series of SAM blocks and MSA blocks that are stacked successively (as presented in Sec. 3.3). SAM and MSA denote the scale-aware modulation module and multi-head self-attention module, respectively.

have focused on the efficiency of CNNs. To further improve the performance of CNNs, attention-based models such as SE-Net [19], Non-local Networks [54], and CBAM [55] have been proposed to enhance the modeling of channel or spatial attention. EfficientNets [45, 46] and MobileNetV3 [17] have employed neural architecture search (NAS) [70] to develop efficient network architectures. ConvNeXt [30] adopts the hierarchical design of Vision Transformers to enhance CNN performance while retaining the simplicity and effectiveness of CNNs. Recently, several studies [13, 16, 60] have utilized convolutional modulation as a replacement for self-attention, resulting in improved performance. Specifically, FocalNet [60] utilizes a stack of depth-wise convolutional layers to encode features across short to long ranges and then injects the modulator into the tokens using an element-wise affine transformation. Conv2Former [16] achieves good recognition performance using a simple 11×11 depth-wise convolution. In contrast, our scale-aware modulation also employs depth-wise convolution as a basic operation but introduces multi-head mixed convolution and scale-aware aggregation.

2.3. Hybrid CNN-Transformer Networks

A popular topic in visual recognition is the development of hybrid CNN-Transformer architectures. Recently, several studies [12, 41, 56, 69] have demonstrated the effectiveness of combining Transformers and convolutions to leverage the strengths of both architectures. CvT [56] first introduced depth-wise and point-wise convolutions before self-attention. CMT [12] proposed a hybrid network that utilizes Transformers to capture long-range dependencies and CNNs to model local features. MobileViT [33], EdgeNeXt [32], MobileFormer [4], and EfficientFormer [24] reintroduced convolutions to Transformers for efficient network design and demonstrated exceptional performance in image classification and downstream applications. However, the current hybrid networks lack the ability to model

range dependency transitions, making it challenging to improve their performance. In this paper, we propose an evolutionary hybrid network that addresses this limitation and showcases its importance.

3. Method

3.1. Overall Architecture

The overall architecture of our proposed Scale-Aware Modulation Transformer (SMT) is illustrated in Fig. 2. The network comprises four stages, each with downsampling rates of $\{4, 8, 16, 32\}$. Instead of constructing an attention-free network, we first adopt our proposed Scale-Aware Modulation (SAM) in the top two stages, followed by a penultimate stage where we sequentially stack one SAM block and one Multi-Head Self-Attention (MSA) block to model the transition from capturing local to global dependencies. For the last stage, we solely use MSA blocks to capture long-range dependencies effectively. For the Feed-Forward Network (FFN) in each block, we adopt the detail-specific feedforward layers as used in Shunted [38].

3.2. Scale-Aware Modulation

Multi-Head Mixed Convolution We propose the Multi-Head Mixed Convolution (MHMC), which introduces multiple convolutions with different kernel sizes, enabling it to capture various spatial features across multiple scales. Furthermore, MHMC can expand the receptive field using a large convolutional kernel, enhancing its ability to model long-range dependencies. As depicted in Fig. 3(b), MHMC partitions input channels into N heads and applies distinct depth-wise separable convolutions to each head, which reduces the parameter size and computational cost. To simplify our design process, we initialize the kernel size with 3×3 and gradually increase it by 2 per head. This approach enables us to regulate the range of receptive fields and multi-granularity information by merely adjusting the
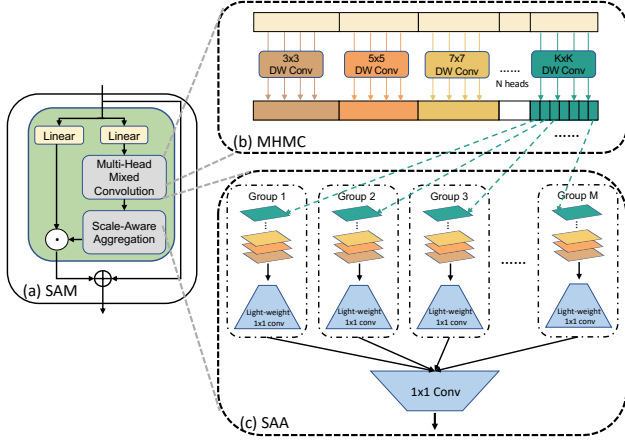
Figure 3: (a) The schematic illustration of the proposed scale-aware modulation (SAM). (b) and (c) are the module descriptions of multi-head mixed convolution (MHMC) and scale-aware aggregation (SAA), respectively.
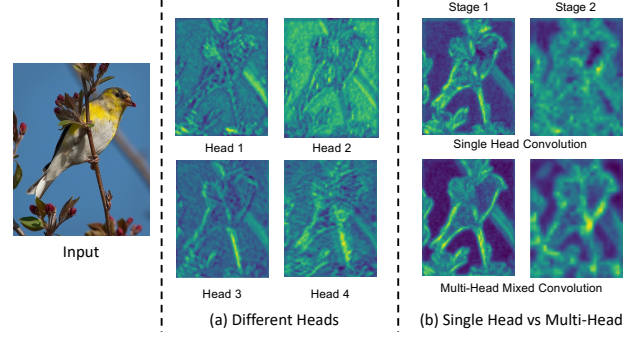

Figure 4: (a) Visualization of the output values of different heads in the MHMC in the first stage. (b) Visualization of the modulation values (corresponding to the left side of $\odot$ in Eq. 3) under single-head and multi-head mixed convolution in the last layer during the top two stages. All maps are upsampled for display.

number of heads. Our proposed MHMC can be formulated as follows:

$$MHMC(X) = Concat(DW_{k_1 \times k_1}(x_1), \ldots, DW_{k_n \times k_n}(x_n)) \quad (1)$$

where $x = [x_1, x_2, ..., x_n]$ means to split up the input feature x into multiple heads in the channel dimension and $k_i \in \{3, 5, \ldots, K\}$ denotes the kernel size increases monotonically by 2 per head.

As shown in Fig. 4(a), each distinct convolution feature map learns to focus on different granularity features in an adaptive manner, as expected. Notably, when we compare the single-head and multi-head by visualizing modulation maps in Fig. 4(b), we find that the visualization under multi-head depicts the foreground and target objects accurately in stage 1, while filtering out background information effectively. Moreover, it can still present the overall shape of the target object as the network becomes deeper, while the information related to the details is lost under the single-head convolution. This indicates that MHMC has the ability to capture local details better than a single head at the shallow stage, while maintaining detailed and semantic information about the target object as the network becomes deeper.

Scale-Aware Aggregation To enhance information interaction across multiple heads in MHMC, we introduce a new lightweight aggregation module, termed Scale-Aware Aggregation (SAA), as shown in Fig. 3(c). The SAA involves an operation that shuffles and groups the features of different granularities produced by the MHMC. Specifically, we select one channel from each head to construct a group, and then we utilize the inverse bottleneck structure to perform an up-down feature fusion operation within each

group, thereby enhancing the diversity of multi-scale features. However, a well-designed grouping strategy enables us to introduce only a small amount of computation while achieving desirable aggregation results. Notably, let the input $X \in \mathbb{R}^{H \times W \times C}$, $Groups = \frac{C}{Heads}$, which means the number of groups is inversely proportional to the number of heads. Subsequently, we perform cross-group information aggregation for all features using point-wise convolution to achieve cross-fertilization of global information. The process of SAA can be formulated as follows:

$$
\begin{aligned}
M &= W_{inter}([G_1, G_2, \ldots, G_M]), \\
G_i &= W_{intra}([H_1^i, H_2^i, \ldots, H_N^i]), \\
H_j^i &= DWConv_{k_j \times k_j}(x_j^i) \in \mathbb{R}^{H \times W \times 1}.
\end{aligned}
\quad (2)
$$

where W_{inter} and W_{intra} are weight matrices of point-wise convolution. $j \in \{1, 2, \ldots, N\}$ and $i \in \{1, 2, \ldots, M\}$, where N and $M = \frac{C}{N}$ denote the number of heads and groups, respectively. Here, $H_j \in \mathbb{R}^{H \times W \times M}$ represents the j-th head with depth-wise convolution, and H_j^i represents the i-th channel in the j-th head.

Fig. 5 shows that our SAA module explicitly strengthens the semantically relevant low-frequency signals and precisely focuses on the most important parts of the target object. For instance, in stage 2, the eyes, head and body are clearly highlighted as essential features of the target object, resulting in significant improvements in classification performance. Compared to the convolution maps before aggregation, our SAA module demonstrates a better ability to capture and represent essential features for visual recognition tasks. (More visualizations can be found in Appendix).

Scale-Aware Modulation As illustrated in Fig. 3(a), after capturing multi-scale spatial features using MHMC and aggregating them with SAA, we obtain an output feature map,
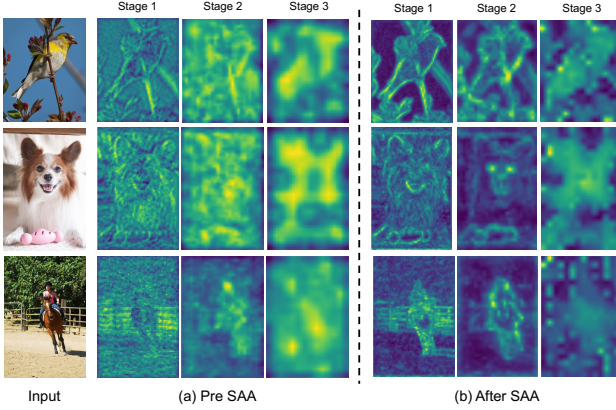
Figure 5: (a) Visualization of the modulation values before SAA. (b) Visualization of the modulation values after SAA.

which we refer to as the modulator M. We then adopt this modulator to modulate the value V using the scalar product. For the input features $X \in \mathbb{R}^{H \times W \times C}$, we compute the output Z as follows:

$$
\begin{aligned}
Z &= M \odot V, \\
V &= W_v X, \\
M &= SAA(MHMC(W_s X)).
\end{aligned}
\tag{3}
$$

where $\odot$ is the element-wise multiplication, W_v and W_s are weight martices of linear layers. Since the modulator is calculated via Eq. 3, it changes dynamically with different inputs, thereby achieving adaptively self-modulation. Moreover, unlike self-attention, which computes an $N \times N$ attention map, the modulator retains the channel dimension. This feature allows for spatial- and channel-specific modulation of the value after element-wise multiplication, while also being memory-efficient, particularly when processing high-resolution images.

3.3. Scale-Aware Modulation Transformer

Evolutionary Hybrid Network In this section, we propose to reallocate the appropriate computational modules according to the variation pattern in the network's capture range dependencies to achieve better computational performance. We propose using MSA blocks only from the penultimate stage to reduce the computational burden. Furthermore, to effectively simulate the transition pattern, we put forth two hybrid stacking strategies for the penultimate stage: (i) sequentially stacking one SAM block and one MSA block, which can be formulated as $(SAM \times 1 + MSA \times 1) \times \frac{N}{2}$, depicted in Fig. 6(i); (ii) using SAM blocks for the first half of the stage and MSA blocks for the second half, which can be formulated as $(SAM \times \frac{N}{2} + MSA \times \frac{N}{2})$, depicted in Fig. 6(ii).

To assess the efficacy of these hybrid stacking strategies, we evaluated their top-1 accuracy on the ImageNet-1K, as
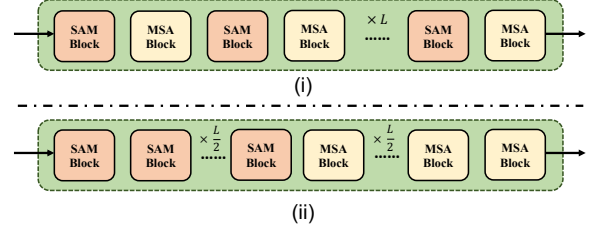

Figure 6: Two proposed hybrid stacking strategies.

shown in Table 9. Moreover, as depicted in Fig. 7, we calculate the relative receptive field of the MSA blocks in the penultimate stage, followed by the approach presented in [3]. It is noteworthy that there is a slight downward trend in the onset of the relative receptive field in the early layers. This decline can be attributed to the impact of the SAM on the early MSA blocks, which emphasize neighboring tokens. We refer to this phenomenon as the adaptation period. As the network becomes deeper, we can see a smooth and steady upward trend in the receptive field, indicating that our proposed evolutionary hybrid network effectively simulates the transition from local to global dependency capture.
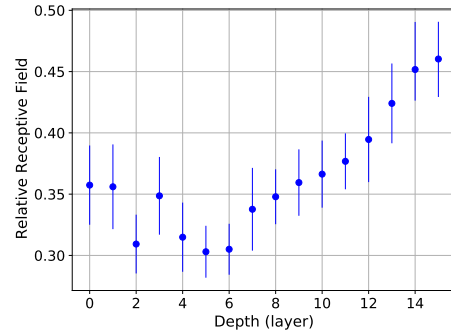
Figure 7: The receptive field of SMT-B's relative attention across depth, with error bars representing standard deviations across various attention heads.

4. Experiments

To ensure a fair comparison under similar parameters and computation costs, we construct a range of SMT variants. We validate our SMTs on ImageNet-1K [8] image classification, MS COCO [27] object detection, and ADE20K [68] semantic segmentation. Besides, extensive ablation studies provide a close look at different components of the SMT. (The detailed model settings are presented in Appendix)

4.1. Image Classification on ImageNet-1K

Setup We conduct an evaluation of our proposed model and compare it with various networks on ImageNet-1K

(a) Tiny Models

method	image size	#param.	FLOPs	ImageNet top-1 acc.
RegNetY-1.6G [35]	224^2	11.2M	1.6G	78.0
EffNet-B3 [45]	300^2	12M	1.8G	81.6
PVTv2-b1 [53]	224^2	13.1M	2.1G	78.7
EfficientFormer-L1 [24]	224^2	12.3M	1.3G	79.2
Shunted-T [38]	224^2	11.5M	2.1G	79.8
Conv2Former-N [16]	224^2	15M	2.2G	81.5
SMT-T(Ours)	224^2	11.5M	2.4G	**82.2**

(b) Small Models

method	image size	#param.	FLOPs	ImageNet top-1 acc.
RegNetY-4G [35]	224^2	21M	4.0G	80.0
EffNet-B4 [45]	380^2	19M	4.2G	82.9
DeiT-S [48]	224^2	22M	4.6G	79.8
Swin-T [29]	224^2	29M	4.5G	81.3
ConvNeXt-T [30]	224^2	29M	4.5G	82.1
PVTv2-b2 [53]	224^2	25.0M	4.0G	82.0
Focal-T [61]	224^2	29.1M	4.9G	82.2
Shunted-S [38]	224^2	22.4M	4.9G	82.9
CMT-S [12]	224^2	25.1M	4.0G	83.5
FocalNet-T [60]	224^2	28.6M	4.5G	82.3
Conv2Former-T [16]	224^2	27M	4.4G	83.2
HorNet-T [37]	224^2	23M	4.0G	83.0
InternImage-T [51]	224^2	30M	5.0G	83.5
MaxViT-T [49]	224^2	31M	5.6G	83.6
SMT-S(Ours)	224^2	20.5M	4.7G	**83.7**

(c) Base Models

method	image size	#param.	FLOPs	ImageNet top-1 acc.
RegNetY-8G [35]	224^2	39M	8.0G	81.7
EffNet-B5 [45]	456^2	30M	9.9G	83.6
Swin-S [29]	224^2	49.6M	8.7G	83.0
CoAtNet-1 [7]	224^2	42M	8.0G	83.3
PVTv2-b4 [53]	224^2	63M	10.0G	83.6
SwinV2-S/8 [28]	256^2	50M	12.0G	83.7
PoolFormer-m36 [63]	224^2	56.2M	8.8G	82.1
Shunted-B [38]	224^2	39.6M	8.1G	84.0
InternImage-S [51]	224^2	50.0M	8.0G	84.2
Conv2Former-S [16]	224^2	50.0M	8.7G	84.1
Swin-B [29]	224^2	87.8M	15.4G	83.4
ConvNeXt-B [30]	224^2	89M	15.4G	83.8
Focal-B [61]	224^2	89.8M	16.4G	83.8
FocalNet-B [60]	224^2	88.7M	15.4G	83.9
HorNet-B [37]	224^2	87M	15.6G	84.2
SMT-B(Ours)	224^2	32.0M	7.7G	**84.3**

Table 1: Comparison of different backbones on ImageNet-1K classification.

ImageNet-22K pre-trained models

method	image size	#param.	FLOPs	ImageNet top-1 acc.
ViT-B/16 [9]	384^2	86.0M	55.4G	84.0
ViT-L/16 [9]	384^2	307.0M	190.7G	85.2
Swin-Large [29]	$224^2/224^2$	196.5M	34.5G	86.3
Swin-Large [29]	$384^2/384^2$	196.5M	104.0G	87.3
FocalNet-Large [60]	$224^2/224^2$	197.1M	34.2G	86.5
FocalNet-Large [60]	$224^2/384^2$	197.1M	100.6G	87.3
InternImage-L [51]	$224^2/384^2$	223M	108G	87.7
InternImage-XL [51]	$224^2/384^2$	335M	163G	88.0
SMT-L(Ours)	$224^2/224^2$	80.5M	17.7G	**87.1**
SMT-L(Ours)	$224^2/384^2$	80.5M	54.6G	**88.1**

Table 2: ImageNet-1K finetuning results with models pretrained on ImageNet-22K. Numbers before and after "/" are resolutions used for pretraining and finetuning, respectively

AdamW optimizer [21, 31] with a weight decay of 0.05 and a learning rate of 1×10^{-3}. In addition, we investigate the effectiveness of SMTs when pretrained on ImageNet-22K. (Further details regarding the training process can be found in Appendix)

Results Tab. 1 presents a comparison of our proposed SMT with various models, and the results demonstrate that our models outperform various architectures with fewer parameters and lower computation costs. Specifically, concerning the tiny-sized model, SMT achieves an impressive top-1 accuracy of 82.2%, surpassing PVTv2-b1 [53] and Shunted-T [38] by significant margins of 3.5% and 2.4%, respectively. Furthermore, when compared to small-sized and base-sized models, SMT maintains its leading position. Notably, SMT-B achieves a top-1 accuracy of 84.3% with only 32M parameters and 7.7GFLOPs of computation, outperforming many larger models such as Swin-B [29], ConvNeXt-B [30], and FocalNet-B [60], which have over 70M parameters and 15GFLOPs of computation. Additionally, to evaluate the scalability of the SMT, we have also created smaller and larger models, and the experimental results are presented in the Appendix.

We also report the ImageNet-22K pre-training results here in Tab. 2. When compared to the previously best results, our models achieve significantly better accuracy with a reduced number of parameters and FLOPs. SMT-L attains an 88.1% top-1 accuracy, surpassing InternImage-XL by 0.1% while utilizing significantly fewer parameters (80.5M vs. 335M) and exhibiting lower FLOPs (54.6G vs. 163G). This highly encouraging outcome underscores the impressive scalability capabilities of SMT.

4.2. Object Detection and Instance Segmentation

Setup We make comparisons on object detection with COCO 2017 [27]. We use SMT-S/B pretrained on

classification [8]. To ensure a fair comparison, we follow the same training recipes as previous works [48, 29, 38]. Specifically, we train the models for 300 epochs with an image size of 224×224 and report the top-1 validation accuracy. The batch size used is 1024, and we employ the

Backbone	Params (M)	FLOPs (G)	Mask R-CNN 1× schedule						Mask R-CNN 3× schedule + MS					
			AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}	AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}
ResNet50 [15]	44.2	260	38.0	58.6	41.4	34.4	55.1	36.7	41.0	61.7	44.9	37.1	58.4	40.1
Twins-SVT-S [5]	44.0	228	43.4	66.0	47.3	40.3	63.2	43.4	46.8	69.2	51.2	42.6	66.3	45.8
Swin-T [29]	47.8	264	42.2	64.6	46.2	39.1	61.6	42.0	46.0	68.2	50.2	41.6	65.1	44.8
PVTv2-B2 [52]	45.0	-	45.3	67.1	49.6	41.2	64.2	44.4	-	-	-	-	-	-
Focal-T [61]	48.8	291	44.8	67.7	49.2	41.0	64.7	44.2	47.2	69.4	51.9	42.7	66.5	45.9
CMT-S [12]	44.5	249	44.6	66.8	48.9	40.7	63.9	43.4	-	-	-	-	-	-
FocalNet-T [60]	48.9	268	46.1	68.2	50.6	41.5	65.1	44.5	48.0	69.7	53.0	42.9	66.5	46.1
SMT-S	**40.0**	265	**47.8**	**69.5**	**52.1**	**43.0**	**66.6**	**46.1**	**49.0**	**70.1**	**53.4**	**43.4**	**67.3**	**46.7**
ResNet101 [15]	63.2	336	40.4	61.1	44.2	36.4	57.7	38.8	42.8	63.2	47.1	38.5	60.1	41.3
Swin-S [29]	69.1	354	44.8	66.6	48.9	40.9	63.4	44.2	48.5	70.2	53.5	43.3	67.3	46.6
Swin-B [29]	107.1	497	46.9	69.2	51.6	42.3	66.0	45.5	48.5	69.8	53.2	43.4	66.8	46.9
Twins-SVT-B [5]	76.3	340	45.2	67.6	49.3	41.5	64.5	44.8	48.0	69.5	52.7	43.0	66.8	46.6
PVTv2-B4 [52]	82.2	-	47.5	68.7	52.0	42.7	66.1	46.1	-	-	-	-	-	-
Focal-S [61]	71.2	401	47.4	69.8	51.9	42.8	66.6	46.1	48.8	70.5	53.6	43.8	67.7	47.2
FocalNet-S [60]	72.3	365	48.3	**70.5**	53.1	43.1	67.4	46.2	49.3	70.7	54.2	43.8	67.9	47.4
SMT-B	**51.7**	**328**	**49.0**	70.2	**53.7**	**44.0**	**67.6**	**47.4**	**49.8**	**71.0**	**54.4**	**44.0**	**68.0**	47.3

Table 3: Object detection and instance segmentation with Mask R-CNN on COCO. Only the 3× schedule has the multi-scale training. All backbones are pre-trained on ImageNet-1K.

Method	Backbones	#Params	FLOPs	AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}
	ResNet50 [15]	82.0M	739G	46.3	64.3	50.5	40.1	61.7	43.4
	Swin-T [29]	86.0M	745G	50.5	69.3	54.9	43.7	66.6	47.1
Cascade [1]	ConvNeXt [30]	-	741G	50.4	69.1	54.8	43.7	66.5	47.3
	Shuffle-T [20]	86.0M	746G	50.8	69.6	55.1	44.1	66.9	48.0
	FocalNet-T [60]	87.1M	751G	51.5	70.3	56.0	-	-	-
	SMT-S	**77.9M**	744G	**51.9**	**70.5**	**56.3**	**44.7**	**67.8**	**48.6**

Method	Backbones	#Params	FLOPs	AP^b	AP^b_{50}	AP^b_{75}	AP_S	AP_M	AP_L
	ResNet50 [15]	37.7M	240G	39.0	58.4	41.8	22.4	42.8	51.6
	Swin-T [29]	38.5M	245G	45.0	65.9	48.4	29.7	48.9	58.1
RetinaNet [26]	Focal-T [61]	39.4M	265G	45.5	66.3	48.8	31.2	49.2	58.7
	Shunted-S [38]	32.1M	-	46.4	66.7	50.4	31.0	51.0	60.8
	SMT-S	**30.1M**	247G	**47.3**	**67.8**	**50.5**	**32.5**	**51.1**	**62.3**

Table 4: COCO detection and segmentation with the **Cascade Mask R-CNN** and **RetinaNet**. The performances are reported on the COCO *val* dataset under the 3× schedule.

Method	Backbone	#Param.	FLOPs	AP^b	AP^b_{50}	AP^b_{75}
	R-50 [15]	106.1M	166G	44.5	63.4	48.2
	Swin-T [29]	109.7M	172G	47.9	67.3	52.3
Sparse R-CNN [42]	Focal-T [61]	110.8M	196G	49.0	69.1	53.2
	FocalNet-T [60]	111.2M	178G	49.9	69.6	54.4
	SMT-S	**102.0M**	171G	**50.2**	**69.8**	**54.7**
	R-50 [15]	32.1M	205G	43.5	61.9	47.0
	Swin-T [29]	35.7M	212G	47.2	66.5	51.3
ATSS [66]	Focal-T [61]	36.8M	239G	49.5	68.8	53.9
	FocalNet-T [60]	37.2M	220G	49.6	68.7	54.5
	SMT-S	**28.0M**	214G	49.9	68.9	54.7
	R-50 [15]	47.7M	244G	49.2	66.7	53.8
	Swin-T [29]	48.2M	252G	51.3	69.0	55.9
DINO [64]	Swin-S [29]	69.5M	332G	53.0	71.2	57.6
	SMT-S	**39.9M**	309G	**54.0**	**71.9**	**59.0**

Table 5: A comparison of models with three different object detection frameworks.

ImageNet-1K as the foundation for three well-known object detectors: Mask R-CNN [14], Cascade Mask R-CNN [1], and RetinaNet [26]. To demonstrate a consistent comparison, two training schedules (1× schedule with 12 epochs and 3× schedule with 36 epochs) are adopted in Mask R-CNN. In 3× schedule, we use a multi-scale training strategy by randomly resizing the shorter side of an image to between [480, 800]. We take AdamW optimizer with a weight decay of 0.05 and an initial learning rate of 2×10^{-4}. Both models are trained with batch size 16. To further showcase the versatility of SMT, we conducted a performance evaluation of SMT with three other prominent object detection frameworks, namely Sparse RCNN [42], ATSS [66], and DINO [64]. We initialize the backbone with weights pretrained on ImageNet-1K and fine-tune the model using a 3× schedule for Sparse RCNN and ATSS.

Results Tab. 3 presents the superior performance of SMT over other networks with Mask R-CNN [14] under various model sizes. Specifically, SMT demonstrates a significant improvement in box mAP of 5.6 and 4.2 over Swin Transformer in 1× schedule under small and base model sizes, respectively. Notably, with 3× schedule and multi-scale training, SMT still consistently outperforms various backbones. For instance segmentation, the results also demonstrate that our SMT achieves higher mask mAP in comparison to previous SOTA networks. In particular, for small and base models in the 1× schedule, we achieve 1.5 and 0.9 points higher than FocalNet, respectively. Furthermore, to assess the generality of SMT, we trained two additional detec-

tion models, Cascade Mask R-CNN [1] and RetinaNet [26], using SMT-S as the backbone. The results, presented in Tab. 4, show clear improvements over various backbones in both box and mask mAPs. The resulting box mAPs for Sparse R-CNN, ATSS and DINO are presented in Tab. 5, which indicate that SMT outperforms other networks consistently across all detection frameworks, highlighting its exceptional performance in downstream tasks.

4.3. Semantic Segmentation on ADE20K

Setup We evaluate the SMT for semantic segmentation using the ADE20K dataset. To conduct the evaluation, we use UperNet as the segmentation method and closely followed the training settings proposed by [29]. Specifically, we train UperNet [57] for 160k iterations with an input resolution of 512×512. We employ the AdamW optimizer with a weight decay of 0.01, and set the learning rate to 6×10^{-5}.

Results The results are presented in Tab. 6, which shows that our SMT outperforms Swin, FocalNet, and Shunted Transformer significantly under all settings. Specifically, SMT-B achieves 1.5 and 0.9 mIoU gains compared to Swin-B and a 0.6 and 0.1 mIoU improvement over Focal-B at single- and multi-scale, respectively, while consuming significantly fewer FLOPs and reducing the model size by more than 50%. Even for the SMT with a small model size, it achieves comparable accuracy with the previous SOTA models which have a larger model size.

Backbone	#Param(M)	FLOPs(G)	$mIoU_{ss}$	$mIoU_{ms}$
ResNet-101 [15]	86	1029	44.9	-
DeiT-S [48]	52	1099	44.0	-
Swin-T [29]	60	941	44.5	45.8
Focal-T [61]	62	998	45.8	47.0
FocalNet-T [61]	61	949	46.8	47.8
Swin-S [29]	81	1038	47.6	49.5
ConvNeXt-S [30]	82	1027	49.6	-
Shunted-S [38]	52	940	48.9	49.9
FocalNet-S [60]	84	1044	49.1	50.1
Focal-S [61]	85	1130	48.0	50.0
Swin-B [29]	121	1188	48.1	49.7
Twins-SVT-L [5]	133	-	48.8	50.2
Focal-B [61]	126	1354	49.0	50.5
SMT-S	50.1	935	49.2	50.2
SMT-B	61.8	1004	**49.6**	**50.6**

Table 6: Semantic segmentation on ADE20K [68]. All models are trained with UperNet [57]. $mIoU_{ms}$ means multi-scale evaluation.

4.4. Ablation Study

Number of heads in Multi-Head Mixed Convolution Table 7 shows the impact of the number of convolution heads in the Multi-Head Mixed Convolution (MHMC) on our model's performance. The experimental results indicate that while increasing the number of diverse convolutional kernels is advantageous for modeling multi-scale features and expanding the receptive field, adding more heads introduces larger convolutions that may negatively affect network inference speed and reduce throughput. Notably, we observed that the top-1 accuracy on ImageNet-1K peaks when the number of heads is 4, and increasing the number of heads does not improve the model's performance. This findings suggest that introducing excessive distinct convolutions or using a single convolution is not suitable for our SMT, emphasizing the importance of choosing the appropriate number of convolution heads to model a specific degree of multi-scale spatial features.

Heads Number	Params(M)	FLOPs(G)	top-1 (%)	throughput (images/s)
1	11.5	2.4	81.8	983
2	11.5	2.4	82.0	923
4	11.5	**2.4**	**82.2**	833
6	11.6	2.5	81.9	766
8	11.6	2.5	82.0	702

Table 7: Model performance with number of heads in MHMC. We analyzed the model's performance for the number of heads ranging from 1 to 8. Throughput is measured using a V100 GPU, following [29].

Different aggregation strategies After applying the MHMC, we introduce an aggregation module to achieve information fusion. Table 8 presents a comparison of different aggregation strategies, including a single linear layer, two linear layers, and an Invert BottleNeck (IBN) [39]. Our proposed scale-aware aggregation (SAA) consistently outperforms the other fusion modules, demonstrating the effectiveness of SAA in modeling multi-scale features with fewer parameters and lower computational costs. Notably, as the size of the model increases, our SAA can exhibit more substantial benefits while utilizing a small number of parameters and low computational resources.

Aggregation Strategy	Params (M)	FLOPs (G)	top-1 (%)
No aggregation	10.9	2.2	81.5
Single Linear ($c \rightarrow c$)	11.2	2.3	81.6
Two Linears ($c \rightarrow c \rightarrow c$)	11.5	2.4	81.9
IBN ($c \rightarrow 2c \rightarrow c$)	12.1	2.6	82.1
SAA($c \rightarrow 2c \rightarrow c$)	11.5	2.4	**82.2**

Table 8: Model performance for different aggregation methods.

Different hybrid stacking strategies In Sec. 3.3, we propose two hybrid stacking strategies to enhance the modeling

of the transition from local to global dependencies. The results shown in Table 9 indicate that the first strategy which sequentially stacks one scale-aware modulation block and one multi-head self-attention block is better, achieving a performance gain of 0.3% compared to the other strategy. Furthermore, the strategy stacking all MSA blocks achieves comparable performance as well, which means retaining the MSA block in the last two stages is crucial.

Stacking Strategy	Hybrid	Params (M)	FLOPs (G)	top-1 (%)
$(SAM \times N)$	✗	11.8	2.5	81.4
$(MSA \times N)$	✗	11.2	2.3	81.8
$(SAM \times 1 + MSA \times 1) \times \frac{N}{2}$	✓	11.5	2.4	**82.2**
$(SAM \times \frac{N}{2} + MSA \times \frac{N}{2})$	✓	11.5	2.4	81.9

Table 9: Top-1 accuracy on ImageNet-1K of different stacking strategies.

Component Analysis   In this section, we investigate the individual contributions of each component by conducting an ablation study on SMT. Initially, we employ a single-head convolution module and no aggregation module to construct the modulation.   Based on this, we build an attention-free network, which can achieve 80% top-1 accuracy on the ImageNet-1K dataset. The effects of all the proposed methods on the model's performance are given in Tab. 10, which can be summarized as followings.

- **Multi-Head Mixed Convolution (MHMC)** To enhance the model's ability to capture multi-scale spatial features and expand its receptive field, we replaced the single-head convolution with our proposed MHMC. This module proves to be effective for modulation, resulting in a 0.8% gain in accuracy.

- **Scale-Aware Aggregation (SAA)** We replace the single linear layer with our proposed scale-aware aggregation. The SAA enables effective aggregation of the multi-scale features captured by MHMC. Building on the previous modification, the replacement leads to a 1.6% increase in performance.

- **Evolutionary Hybrid Network (EHN)** We incorporate the self-attention module in the last two stages of our model, while also implementing our proposed hybrid stacking strategy in the penultimate stage, which improves the modeling of the transition from local to global dependencies as the network becomes deeper, resulting in a significant gain of 2.2% in performance based on the aforementioned modifications.

MHMC	SAA	EHN	Params(M)	FLOPs(G)	top-1 (%)
			11.1	2.3	80.0 (↑0.0)
✓			11.2	2.3	80.8 (↑0.8)
✓	✓		12.1	2.5	81.6 (↑1.6)
✓	✓	✓	11.5	2.4	82.2 (↑2.2)

Table 10: Component analysis for SMT. Three variations are gradually added to the original attention-free network.

5. Conclusion

In this paper, we introduce a new hybrid ConvNet and vision Transformer backbone, namely Scale-Aware Modulation Transformer (SMT), which can effectively simulate the transition from local to global dependencies as the network becomes deeper, resulting in superior performance. To satisfy the requirement of foundation models, we propose a new Scale-Aware Modulation that includes a potent multi-head mixed convolution module and a lightweight scale-aware aggregation module. Extensive experiments demonstrate the efficacy of SMT as a backbone for various downstream tasks, achieving comparable or better performance than well-designed ConvNets and vision Transformers, with fewer parameters and FLOPs. We anticipate that the exceptional performance of SMT on diverse vision problems will encourage its adoption as a promising new generic backbone for efficient visual modeling.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[2] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 357–366, 2021.

[3] Wuyang Chen, Xianzhi Du, Fan Yang, Lucas Beyer, Xiaohua Zhai, Tsung-Yi Lin, Huizhong Chen, Jing Li, Xiaodan Song, Zhangyang Wang, et al. A simple single-scale vision transformer for object detection and instance segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*, pages 711–727. Springer, 2022.

[4] Yinpeng Chen, Xiyang Dai, Dongdong Chen, Mengchen Liu, Xiaoyi Dong, Lu Yuan, and Zicheng Liu. Mobile-former: Bridging mobilenet and transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5270–5279, 2022.

[5] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34:9355–9366, 2021.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Scharwächter, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset. In *CVPR Workshop on the Future of Datasets in Vision*, volume 2, 2015.

[7] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *Advances in Neural Information Processing Systems*, 34:3965–3977, 2021.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

[10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[11] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE TPAMI*, 43(2):652–662, 2021.

[12] Jianyuan Guo, Kai Han, Han Wu, Yehui Tang, Xinghao Chen, Yunhe Wang, and Chang Xu. Cmt: Convolutional neural networks meet vision transformers (supplementary mate-

rial. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12175–12185, 2022.

[13] Meng-Hao Guo, Cheng-Ze Lu, Zheng-Ning Liu, Ming-Ming Cheng, and Shi-Min Hu. Visual attention network. *arXiv preprint arXiv:2202.09741*, 2022.

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[16] Qibin Hou, Cheng-Ze Lu, Ming-Ming Cheng, and Jiashi Feng. Conv2former: A simple transformer-style convnet for visual recognition. *arXiv preprint arXiv:2211.11943*, 2022.

[17] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.

[18] 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 preprint arXiv:1704.04861*, 2017.

[19] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[20] Zilong Huang, Youcheng Ben, Guozhong Luo, Pei Cheng, Gang Yu, and Bin Fu. Shuffle transformer: Rethinking spatial shuffle for vision transformer. *arXiv preprint arXiv:2106.03650*, 2021.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. 2015.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[23] Jiashi Li, Xin Xia, Wei Li, Huixia Li, Xing Wang, Xuefeng Xiao, Rui Wang, Min Zheng, and Xin Pan. Nextvit: Next generation vision transformer for efficient deployment in realistic industrial scenarios. *arXiv preprint arXiv:2207.05501*, 2022.

[24] Yanyu Li, Geng Yuan, Yang Wen, Eric Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, and Jian Ren. Efficientformer: Vision transformers at mobilenet speed. In *Advances in Neural Information Processing Systems*.

[25] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

[26] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[27] Tsung-Yi Lin, Michael Maire, Serge 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*, pages 740–755. Springer, 2014.

[28] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022.

[29] 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. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[30] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

[32] Muhammad Maaz, Abdelrahman Shaker, Hisham Cholakkal, Salman Khan, Syed Waqas Zamir, Rao Muhammad Anwer, and Fahad Shahbaz Khan. Edgenext: Efficiently amalgamated cnn-transformer architecture for mobile vision applications. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII*, pages 3–20. Springer, 2023.

[33] Sachin Mehta and Mohammad Rastegari. Mobilevit: Lightweight, general-purpose, and mobile-friendly vision transformer. In *International Conference on Learning Representations*.

[34] Lingchen Meng, Hengduo Li, Bor-Chun Chen, Shiyi Lan, Zuxuan Wu, Yu-Gang Jiang, and Ser-Nam Lim. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12309–12318, 2022.

[35] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10428–10436, 2020.

[36] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021.

[37] Yongming Rao, Wenliang Zhao, Yansong Tang, Jie Zhou, Ser-Nam Lim, and Jiwen Lu. Hornet: Efficient high-order spatial interactions with recursive gated convolutions. In *Advances in Neural Information Processing Systems*.

[38] Sucheng Ren, Daquan Zhou, Shengfeng He, Jiashi Feng, and Xinchao Wang. Shunted self-attention via multi-scale token aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10853–10862, 2022.

[39] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[40] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. 2015.

[41] Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. Bottleneck transformers for visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16519–16529, 2021.

[42] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14454–14463, 2021.

[43] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[44] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

[45] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[46] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In *International conference on machine learning*, pages 10096–10106. PMLR, 2021.

[47] Mingxing Tan and Quoc V Le. Mixconv: Mixed depthwise convolutional kernels. In *the 30th British Machine Vision Conference (BMVC) 2019*, 2019.

[48] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[49] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. In *European conference on computer vision*, pages 459–479. Springer, 2022.

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[51] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. *arXiv preprint arXiv:2211.05778*, 2022.

[52] 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. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021.

[53] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022.

[54] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[55] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[56] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22–31, 2021.

[57] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018.

[58] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, pages 1492–1500, 2017.

[59] Weijian Xu, Yifan Xu, Tyler Chang, and Zhuowen Tu. Co-scale conv-attentional image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9981–9990, 2021.

[60] Jianwei Yang, Chunyuan Li, Xiyang Dai, and Jianfeng Gao. Focal modulation networks. In *Advances in Neural Information Processing Systems*.

[61] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. *Advances in Neural Information Processing Systems*, 2021.

[62] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10809–10818, 2022.

[63] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10819–10829, 2022.

[64] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*, 2022.

[65] Pengchuan Zhang, Xiyang Dai, Jianwei Yang, Bin Xiao, Lu Yuan, Lei Zhang, and Jianfeng Gao. Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2998–3008, 2021.

[66] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020.

[67] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.

[68] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

[69] Daquan Zhou, Zhiding Yu, Enze Xie, Chaowei Xiao, Animashree Anandkumar, Jiashi Feng, and Jose M Alvarez. Understanding the robustness in vision transformers. In *International Conference on Machine Learning*, pages 27378–27394. PMLR, 2022.

[70] Barret Zoph and Quoc Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*.