Doubly-Fused ViT: Fuse Information from Vision Transformer Doubly with Local Representation

Li Gao, Dong Nie, Bo Li, and Xiaofeng Ren

Alibaba Group
{liangliang.gl,dong.nie,shize.lb,x.ren}@alibaba-inc.com

Abstract. Vision Transformer (ViT) has recently emerged as a new paradigm for computer vision tasks, but is not as efficient as convolutional neural networks (CNN). In this paper, we propose an efficient ViT architecture, named Doubly-Fused ViT (DFvT), where we feed low-resolution feature maps to self-attention (SA) to achieve larger context with efficiency (by moving downsampling prior to SA), and enhance it with fine-detailed spatial information. SA is a powerful mechanism that extracts rich context information, thus could and should operate at a low spatial resolution. To make up for the loss of details, convolutions are fused into the main ViT pipeline, without incurring high computational costs. In particular, a Context Module (CM), consisting of fused downsampling operator and subsequent SA, is introduced to effectively capture global features with high efficiency. A Spatial Module (SM) is proposed to preserve fine-grained spatial information. To fuse the heterogeneous features, we specially design a Dual AtteNtion Enhancement (DANE) module to selectively fuse low-level and high-level features. Experiments demonstrate that DFvT achieves state-of-the-art accuracy with much higher efficiency across a spectrum of different model sizes. Ablation study validates the effectiveness of our designed components.

Keywords: Vision Transformer, Convolutional Neural Networks, Efficient Network

1 Introduction

For quite some time now, convolutional neural networks (CNN) [24, 43, 45, 29] have dominated computer vision (CV) tasks, such as image classification, object detection, semantic segmentation, and tracking. CNN extracts information hierarchically, and high-level feature representations are obtained by gradually processing features from the bottom to top layers. In addition, using weight sharing and pooling, CNN has the nice property of (approximate) shift-invariance and equivariance. Nevertheless, the CNN architecture has its drawbacks. A convolution kernel is localized and has a fixed size, so local information is efficiently

Code is available at https://github.com/ginobilinie/DFvT

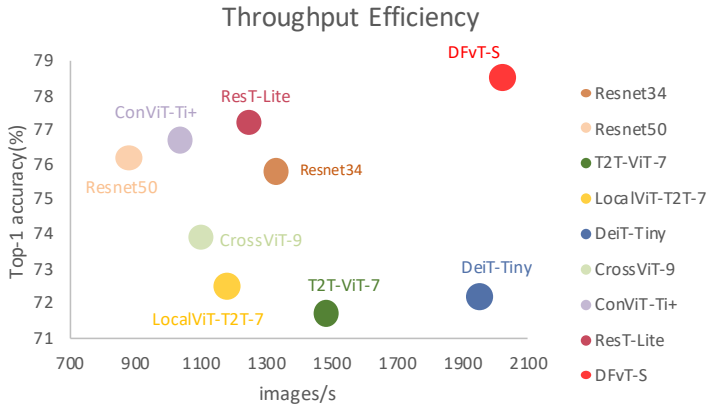

Fig. 1. Scatterplot of top-1 accuracy on ImageNet-1K validation set with respect to speed. The proposed Doubly-Fused ViT (DFvT) achieves state-of-the-art performance with high efficiency, outperforming popular convolution and transformer backbones.

captured, but large receptive fields and long-range dependencies can only be represented by either increasing the depth of the network or utilizing large kernels, both with much higher computation cost. Also, the weights of the convolution kernels are fixed when training is over, and the filter weights cannot be adjusted when the input changes.

Transformer [50] was first proposed for Natural Language Processing (NLP) [10, 23, 61], showing to be superior in performance on machine translation tasks. Instead of localized convolution, transformer uses self-attention mechanisms to capture global contextual information and establishes long-range dependencies, proving a powerful paradigm for feature extraction. One pioneering work of transformer for computer vision tasks is ViT [11], which applies the encoder in the standard transformer to visual tasks by dividing an input image into patches and making an analogy between patches and tokens in NLP.

The ViT design and its follow-up show great promises in achieving higher performance on a variety of vision tasks, but they have disadvantages. Firstly, the patch stem of ViT greatly reduces the resolution of the input image and is not friendly to downstream tasks that require dense pixel prediction, such as semantic segmentation. Secondly, the computational complexity of the vanilla ViT is quadratic in the number of patches/tokens, which leads to a high amount of computation. Many researches has been undertaken to alleviate these two issues. Some efforts [25, 34, 15, 64, 44] introduced hierarchical constructions (commonly used in CNN) to build hierarchical transformers. Others [13, 57, 14] incorporated CNN to add inductive bias to transformers to help improve performance. There were also studies on reducing the computational costs of transformers [49, 4, 12, 32], such as decreasing the number of tokens or modifying the self-attention

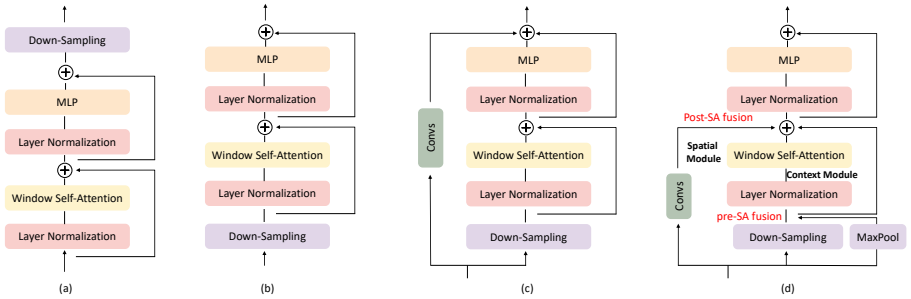

Fig. 2. Different transformer designs (we use a pyramid ViT with window SA as baseline.). (a) Baseline ViT (e.g., [28]). Token maps are (spatially) down-sampled at the end. (b) Downsampled ViT: placing down-sampling at the beginning, before the main transformer block. Computation and memory costs are greatly reduced, but there is a loss of information (and lower accuracy). (c) Parallel ViT: one way to compensate for information loss is to supplement transformer with a separate CNN path (e.g. [9]). Note that the input of CNN and transformer has the same resolution and dimension, which is computation demanding and memory consuming. (d) We propose Doubly-Fused ViT (DFvT), which opens up the transformer block and enhance it with convolution, both before and after self-attention. This design tightly integrates transformer and convolution with CM and SM complementing each other, achieving high accuracy and high speed. See ablation studies.

mechanism. So far, we do not yet have good designs that can make transformers achieves both high performance and high efficiency.

In this work, we propose a simple yet novel vision transformer architecture, named Doubly-Fused ViT (DFvT), which is efficient in both speed and memory comparable to CNN, while retaining ViT's high performance. This is achieved by opening up and redesigning the transformer block and tightly integrating it with convolutions. Specifically, our first change is proposing a Context Module (CM), which moves the downsampling operation (e.g., convolution with stride 2), usually at the end of the transformer block (in a hierarchical design e.g. [28]), to the beginning of the block. Our intuition is that self-attention in the transformer is powerful at capturing large contexts, and this can be done with a low spatial resolution. As a result, computation and memory costs of the transformer are substantially reduced. However, this downsampling operation in CM does result in a serious loss of information, that of spatial details, and a much lower performance. Our second change is to enhance the transformer with detailed spatial information through convolution, with two fusions at distinctive locations, both prior to and after self-attention (SA). The first fusion is in the CM to provide diversified and salient information as input to self-attention; for the second fusion, we introduce a Spatial Module (SM) to provide local details that can be easily missed by the downsampled self-attention which mainly focus on global information. Considering the heterogeneous nature of the two information paths, we

designed a Dual AtteNtion Enhancement (DANE) module to fuse the features. It turns out that our design is effective: DFvT can achieve competitive accuracy with compelling efficiency, validated at multiple model sizes. A plot of speed-accuracy on the small version DFvT is shown in Fig. 1. Several design choices are outlined in Fig. 2, and their performances are compared in the ablation studies.

The contributions of our work are as follows:

- We proposed the Doubly-Fused ViT (DFvT), a general vision transformer backbone with high performance (comparable to standard ViT) and high efficiency (comparable to CNN). We regroup the downsampling operators and self-attention to formulate a Context Module (CM) which can efficiently and effectively capture global information. We also design a Spatial Module (SM) to preserve local details. The context features and fine details are fused with a DANE module.
- A Dual AtteNtion Enhancement (DANE) module is carefully designed to fuse spatial details and contextual features. In DANE, channel attention is adopted to cope with contextual features since no channel interaction in self-attention, and spatial attention is dedicated to local features. Then a automatically selective mechanism is introduced to finally fuse the two heterogeneous features so that we can capture features at different scales (for instance, local and contextual).
- The DFvT design can be instantiated at multiple model sizes (at the level of ResNet101, ResNet50-ResNet18, and MobileNet). We carry out a series of experiments to show that DFvT is flexible and can apply to varying computational demands, outperforming the state-of-the-art (of both transformer and CNN).

2 Related Work

2.1 CNNs

CNNs have achieved great success and dominated computer vision in the past decade [22, 43, 45, 40, 30, 39, 16]. The basic building block in CNNs is a standard convolutional layer, which does well in capturing local details but not in modeling long-range dependency due to limited receptive field. The network requires sufficient context information to perform strong recognition. Stacking convolutional layers is one way to learn context information. With batch normalization (BN) [42] and residual block [16], modern CNNs can go through as deep as 1,000 layers and achieve SoTA performance on many vision tasks. Efforts have been made to improve context modeling in CNN. [35] explored the role of large kernels in segmentation and concluded that large kernels work better than stacking small filters. [63] utilized dilated convolution to aggregate multi-scale features. [8] presented a novel convolution operation with learnable offsets to model long-range dependency and geometric shapes. [55] employed non-local blocks (i.e., self-attention) to increase the receptive field and learn better context.

2.2 Vision Transformer

The pioneering vision transformer (ViT) [11] has recently achieved competitive performance to CNNs, especially when using a large amount of data, which demonstrates the capability and potential of transformers in computer vision. Follow-up research to improve ViT can be roughly categorized into three directions. One is to improve building components of the vision transformer [65, 58, 48, 20, 2] under the isotropic structure (i.e., fixed token numbers and channels) like ViT, for example: T2T-ViT [65] developed a Tokens-to-Token (T2T) transformation to embed local structure for each token instead of using naive tokenization. CaiT [48] proposed a layer-scalar for training a deeper network to achieve better performance, and LV-ViT [20] improved the model training by applying CutMix [66]. CrossViT [2] proposed a dual-path architecture (each with a different scale) to learn multi-scale features. A second direction for improving vision transformer is to introduce pyramid structure [17, 1, 67, 53, 6]. PVT [53] and PiT [17] introduced the pyramid structure which is standard in most CNN models, making PVT and PiT more suitable for image recognition tasks due to the multi-scale features. Swin [28], ViL [67] and Twins [7] further constrained self-attention into a local region and then proposed different strategies to allow information interactions among local regions, leading to even higher recognition accuracy and less computational complexity. RegionViT [3] employed a novel regional-to-local attention within the pyramid structure to boost information communication. The third direction is to combine convolution with transformer. DeiT [47] proposed an efficient training scheme that allows vision transformer to achieve competitive performance with CNN models while training only on ImageNet-1K. LocalViT [26] and ConT [60] presented methods to mix convolutions with self-attention to encode locality information.

2.3 Efficient Architecture for CNNs and ViTs

Computational efficiency is critical for large-scale training, reducing cost, and deployment on edge devices. It is common to reduce the model size by redesigning the model architecture. In CNNs, scaling dimensions of depth/width/resolution [46] and designing efficient operations (e.g., separable convolution [41] and shuffle block [31]) are the widely adopted strategies to build efficient networks. In ViTs, pyramid structure (e.g. [28]) and token sparsification (e.g. [38]) are common ways to design efficient models. Besides, carefully combining CNNs and ViTs can also improve efficiency. For instance, ConViT [9] introduced a form of positional self-attention to control the balance between content-based self-attention and the convolutionally initialized positional self-attention. CvT [57] utilized convolution for token embedding and designed the convolutional transformer block in each stage to bring desirable properties of CNN to ViT and obtained a more efficient model than plain ViT. Mobile-Former [5] utilized a parallel structure with a two-way bridge in-between to combine MobileNet and Transformer, endowing the model with local processing and global interaction capabilities. MobileViT [32], another lightweight model with transformer, introduced transformer

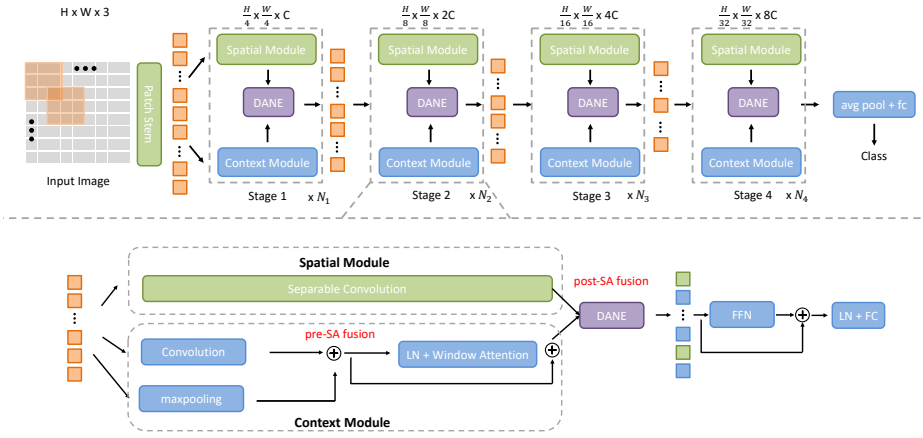

Fig. 3. The pipeline of the proposed Doubly-Fused ViT (DFvT) for image classification, where the number of tokens is reduced as the network goes deeper with a hierarchical representation. We design our efficient transformer block (Spatial Module (SM), Context Module (CM) and DANE). In the CM, we place the downsampling prior to SA to reach larger context information and reduce computation and memory costs simultaneously. Note we use convolved features to make up for the serious loss of information in the CM. To further compensate for the loss of information, we design the SM to retain local details. Moreover, we design a Dual AtteNtion (DANE) module fusing local details and global contexts.

as convolution to rebuild MobileNet, achieving a lightweight and low-latency network on vision tasks. These designs are efficient, but they typically result in a loss of performance, trading accuracy for efficiency.

3 Method

The strength of transformer lies in its multi-head self-attention, which can effectively capture context information (i.e., long-range information dependency modeling) from shallow layers [37]. However, self-attention has a high computational cost which limits the efficiencies of the ViTs. Many works are conducted to reduce the computational complexity in a theoretic aspect, e.g., from quadratic to linear, however, performance drop usually comes together with them. Also, these approaches cannot decrease the memory cost. We propose a simple yet effective method, that is, we reduce the size of the features fed into self-attention to achieve more context with high efficiency, set up a spatial path to retain the detailed information, and specially design a module to fuse the features. The overall framework of the proposed DFvT is presented in Fig. 3. Given an RGB image with a shape of $H \times W \times 3$, it is first encoded into overlapped patches by two consecutive convolutions. The patches can be viewed as tokens in NLP. Unlike

the patch stem in ViT [11], which uses large-kernel plus large-stride convolutions to extract features, we adopt traditional convolution stem (two sequential convolutions with the kernel of 3×3 to strike a balance between inductive biases and the representation learning ability of the following transformer blocks [59]). Then several layers of a redesigned transformer block, tightly enhanced by convolution, are applied to these patches to extract higher-dimensional features. In this work, we adopt the hierarchical modeling approach following [28], so the number of tokens is reduced and the dimension is increased as the layers go deeper. After the last block, the tokens are fed into a global average pooling layer and a fully connected layer to produce the prediction maps.

As shown in Fig. 3, there are four stages in our pipeline, and each stage consists of three modules: a Context Module (CM) to capture large context, a Spatial Module (SM) to retain local details, and a Dual AtteNtion Enhancement module (DANE) to fuse features from the SM and CM. It can be seen that transformer and convolution blocks are tightly integrated, and both fusion steps take place before transformer's FFN step.

3.1 Context Module

We first adopt a convolution block for fast downsampling to obtain high-level semantic context information with three consecutive convolutions. Following the design in ResNet [16], there are a 1×1 convolution, a 3×3 separable convolution with stride 2 for downsampling, and a 1×1 projection convolution to integrate channel information. The convolution kernels extract low-resolution features and allow the following steps to have larger respective field.

Then we adopt N window-based self-attention modules following Swin Transformer [28], which are responsible for extracting medium- and high-level (and long-distance) information. Note that the tokens fed into transformer are downsampled to reduce computational and memory costs and expand the receptive field.

The Context Module (CM) is designed to efficiently and effectively learn context information. The CM downsamples the feature maps and then feed them into self-attention, which ensures the self-attention mechanism has much smaller computational (also memory) costs, and can obtain larger receptive field to enlarge the context.

Pre-SA Fusion We downsample the input feature map of the CM before computing W-MHSA to reduce the amount of computation (approximately a factor of 4 for window-based SA, 1/16 for regular self-attention). However, this reduction in resolution would result in a loss of information, especially salient features. We propose to compensate for this loss in a parameter-free way (i.e., maxpooling) before it is processed by self-attention. Features from the maxpooling steps could be used for such enrichment, as those features are notable ones and can provide translation/rotation for the following transformer.

To this end, we add the feature map from the maxpooling step to that of the convolutions in the transformer. The pre-SA fusion and the output of the W-MHSA in CM can be expressed as:

$$\begin{aligned}\mathcal{F}_{CM} = W\text{-}MHSA(LN(maxpool(\mathcal{F}) + Conv(\mathcal{F}))) \\ + maxpool(\mathcal{F}) + Conv(\mathcal{F})\end{aligned} \tag{1}$$

where $\mathcal{F}$ represents the input feature maps to both CM and SM, and $Conv$ denotes three consecutive convolutions, respectively.

3.2 Spatial Module

It has been observed that preserving the detailed spatial information is crucial to high performance in recognition tasks [51, 62, 33]. However, the CM has seriously lost the fine-detailed features. To compensate the local details, we propose a spatial module (SM) which maintains the large spatial size of tokens. In particular, we adopt separable convolution with a 3×3 kernel to encode the local features. Since the spatial module does not involve any global operator, the spatial details are well preserved in this path.

3.3 DANE

The features of CM and SM represent different scales of information, i.e., SM means local features and CM represents global contexts, it is not a good idea to just simply add the heterogeneous features up. Instead, we design a fusion module called Dual AtteNtion Enhancement (DANE), which consists of a channel attention, a spatial attention and a automatic selective mechanism to allocate attentions to enhance context and spatial features respectively. The structure of the proposed module is shown in Fig. 4.

Since the features extracted by CM lack interaction between channels (self-attention has no inter-channel integration), we perform the channel-wise attention to enhance the useful features of the CM and suppress noise information. This generates a set of global dependencies on channel dimension by aggregating the feature maps in its spatial dimension. Let the token map output from the CM be $\mathcal{F}_{CM}$, we use a global average pooling operation, and two consecutive FC layers for squeeze and excitation [19], to get the channel-wise weights as described below:

$$\mathcal{W}_c = \boldsymbol{f}_{ex}(\boldsymbol{f}_{sq}(avgpool(\mathcal{F}_{CM}))) \tag{2}$$

The feature maps of the SM contain rich fine-grained spatial information, but lack global context information, so we use spatial-wise attention on SM features to learn spatial weighting, enhancing useful spatial areas and suppressing irrelevant ones. Let the SM output be $\mathcal{F}_{SM}$, the spatial-wise weights are generated by aggregating the feature maps in the channel dimension with a FC layer as follows:

$$\mathcal{W}_s = \boldsymbol{f}_{sq}(\mathcal{F}_{SM}) \tag{3}$$

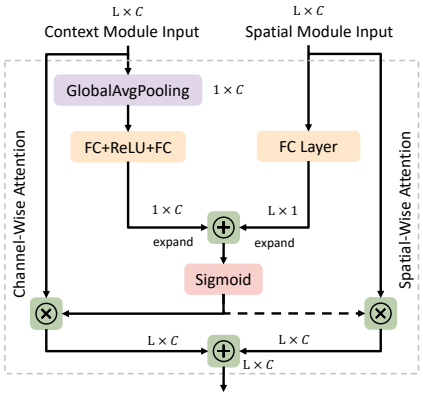

Fig. 4. The framework of the proposed DANE module, which fuses the information from SM and CM with both spatial and channel attention.

With the weight parameters learned, the tokens can learn high- and low-level features from both SM and CM simultaneously, which are calculated as:

$$\begin{aligned} \mathcal{W} &= sigmoid(\mathcal{W}_c + \mathcal{W}_s), \\ \mathcal{F} &= \mathcal{F}_{CM} * \mathcal{W} + \mathcal{F}_{SM} * (1 - \mathcal{W}) \end{aligned} \tag{4}$$

After DANE being introduced, SM and CM are complementary to each other for higher performance. That is, each token in the feature map can obtain both high-level context information and low-level spatial information by adaptively adjusting the weight parameters balancing between SM and CM (sum to 1). This module introduces a small number of parameters but can effectively fuse different level of information representation from these paths.

With fully fusing information from SM and CM, the final output of the transformer block is:

$$\mathcal{F} = MLP(LN(\mathcal{F})) + \mathcal{F} \tag{5}$$

Finally, a standard layernorm operation and a fully connected layer are applied to the feature maps to increase the dimension. After all the basic blocks, the tokens go through global average pooling and a fully connected layer to obtain final predictions.

3.4 Model Scaling

By simply adjusting the dimension and the number of transformer blocks (also with minor adjustment of some other configurations), we can adapt our design to a range of model complexity (and computational cost), namely: Tiny (0.3 GFLOPs), Small (0.8 GFLOPs), and Base (2.5 GFLOPs). These models roughly correspond to MobileNet, ResNet and Swin-T, respectively. Their peformance

Results on ImageNet-1K validation set							
Group	Model	Image Size	Params(M)	FLOPs(G)	Memory(GB)	throughput(images/s)	Top-1 Acc.(%)
0.2G FLOPs and More	MobileNetV1[18]	224	4.2	0.6	5.0	3247.5	70.6
	MobileNetV2[41]	224	3.5	0.3	4.4	2780.6	72.0
	ShffleNetV1 1.5x [69]	224	3.4	0.3	3.3	3140.3	71.6
	ShffleNetV2 1.0x [31]	224	2.3	0.2	2.4	7285.2	69.4
	MobileFormer-52M [5]	224	3.5	0.6	1.8	3033.8	68.7
	PvTv2-B0[52]	224	3.4	0.6	2.5	1624.3	70.5
	DFvT-T	224	4.0	0.3	1.5	4760.1	**73.0**
0.8G FLOPs and More	ResNet18[16]	224	11.7	1.8	1.0	2506.4	69.8
	ResNet34[16]	224	21.8	3.7	1.5	1329.4	75.8
	ResNet50[16]	224	25.6	4.1	3.2	879.0	76.2
	MobileFormer-294M [5]	224	11.8	3.2	6.2	857.7	77.9
	T2T-ViT-7[65]	224	4.3	1.2	3.0	1483.4	71.7
	LocalViT-T[26]	224	5.3	1.3	4.2	1180.3	72.5
	DeiT-Tiny[47]	224	5.7	1.2	2.2	1950.7	72.2
	CrossViT-9[54]	224	8.6	1.8	3.7	1098.9	73.9
	PVT-Tiny[53]	224	13.2	1.9	4.3	1087.8	75.1
	ConViT-Ti+[9]	224	10.0	2.0	4.0	1034.4	76.7
	ResT-Lite[68]	224	10.5	1.4	3.4	1246.3	77.2
	DFvT-S	224	11.2	0.8	2.8	2202.3	**78.3**
2.5G FLOPs and More	ResNet101[16]	224	44.5	7.8	4.7	502.6	77.4
	RegNetX-4G[36]	224	22.1	4.0	7.7	789.8	78.6
	ViT-Base[11]	384	86.8	17.6	OOM	235.4	77.9
	DeiT-Small[47]	224	22.1	4.3	4.8	786.6	79.8
	Swin-Tiny[28]	224	28.3	4.5	8.0	536.8	81.3
	CrossViT-S[54]	224	26.7	5.6	7.0	533.9	81.0
	PVT-Small[53]	224	24.5	3.8	7.0	594.6	79.8
	PVT-Medium[53]	224	44.2	6.7	9.5	390.4	81.2
	Conformer-Ti[13]	224	23.5	5.2	7.1	515.1	81.3
	CvT-13[57]	224	20.0	4.5	6.6	541.9	81.6
	T2T-ViT-14[65]	224	21.5	5.2	6.8	610.7	81.5
	CoTNet-50[27]	224	22.2	3.3	OOM	633.3	81.3
	ConViT-S[9]	224	27.8	5.4	7.9	462.0	81.3
	ResT-Base[68]	224	30.3	4.3	6.4	598.5	81.6
	DFvT-B	224	37.3	2.5	5.5	962.8	**82.0**

Table 1. Comparison of the model family of DFvT and state-of-the-art methods on the ImageNet-1K validation set. The memory cost is tested with a batch size 64, and when testing fps we turn up the batch size to the maximum on a single 2080 Ti GPU. For fair comparison, we don't adopt the mixed precision, and the results reported here are tested on the same platform.

on ImageNet-1K are shown in Sec. 4.1. Details of these models are listed in the supplementary material.

4 Experiments

We conduct experiments on image classification to evaluate and validate performance and efficiency of the proposed DFvT design.

4.1 Image Classification on ImageNet-1K

For the task of image classification, DFvT is trained on ImageNet-1K [22] training set, which contains 1.28 million images of 1k classes, and is tested on the validation set that includes 50k images. In the experiments, image size is set to 224×224, and top-1 accuracy on a single crop is reported. Efficiency is measured using FLOPs, actual throughput (images/sec), and memory usage. We

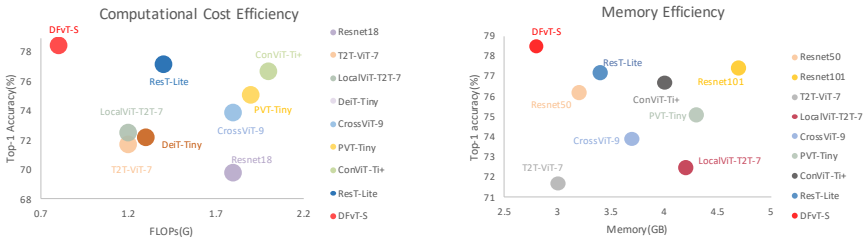

Fig. 5. Illustrations of top-1 accuracy in ImageNet-1K validation set with respect to memory and computational cost.

use AdamW [21] as the optimizer with a cosine decay learning rate scheduler for 300 epochs and linear warm-up for 20 epochs. We set the batch size to 1024 for DFvT-T and DFvT-S, 512 for DFvT-B, the initial learning rate to 1e-3, weight decay to 0.05. The data augmentation and regularization strategies follow [28]. The models are trained on 4 NVIDIA GeForce RTX 2080 Ti GPUs implemented using PyTorch.

The results are shown in Table 1. Our model, DFvT-S achieves 78.3% top-1 accuracy, which is higher than ResNet101 (77.4%) [16] in terms of accuracy, and is approximately 4 times faster with 90% less computational complexity (in FLOPs). Our model also costs 35% less memory. DFvT-S can even compare to small ResNet (i.e., ResNet18) in terms of computational cost and inference throughput, but our model provides a much better accuracy (i.e., 8.5 % higher for top-1 accuracy). Compared to the widely adopted ResNet50, DFvT-S outperforms it in all perspectives.

DFvT-S is also competitive with other ViTs. In the group of 0.8G FLOPs and more, our DFvT-S is among the top performers with respect to the accuracy, computational cost, and inference speed. For example, DFvT-S outperforms many recently developed ViT models, such as T2T-ViT-7, LocalVit-7, CrossViT-9, PVT-Tiny, ConViT-Ti+, and ResT-Lite, concerning the accuracy, FLOPs, inference throughput, and memory. As for MobileFormer-294M, our model achieves comparable accuracy with substantially lower memory and computational costs and higher throughput.

Model Scaling. Table 1 shows the experimental results of our scaled models on image classification (ImageNet-1K). Our tiny model (i.e., DFvT-T) can achieve a competitive accuracy (73.0%) compared to the SOTA lightweight models. More importantly, DFvT-T has advantages over the SOTA methods in terms of the FLOPs, memory costs as well as inference throughput except for ShuffleNetV2 (but note DFvT-T accuracy is 3.6% higher than ShuffleNetV2). For instance, the throughput of DFvT-T is approximately twice as much as MobileNetV2 [41] on a single NVIDIA 2080 Ti GPU while only using one-third of memory. As for the DFvT-B (large model), we also achieve competitive results with lower computational and memory costs. The scatterplots on computational/memory costs and accuracy are shown in Fig. 5.

4.2 Ablation Study

In this section, we conduct extensive ablation experiments to demonstrate the overall effectiveness of our design, the necessity of Context Module (CM), Spatial Module(SM) and DANE block.

Transformer Design Table 2 presents the four options for model design depicted in Fig. 2. It can be observed that baseline ViT has presented decent top-1 accuracy with a small number of parameters, but the computational cost and inference speed are not ideal. If we directly downsampled the input size of the input feature map (Downsampled ViT), we can largely improve the throughput (2 times faster) and greatly reduce the computational complexity (2 times smaller) as well as the memory cost (45% smaller). However, the accuracy becomes much lower, which is assumed due to the information loss. The parallel ViT, which sets up a parallel convolution path to provide local information, can substantially improve the accuracy compared to Downsampled ViT (71.9% to 76.9%). Nevertheless, memory and inference speed become much worse. Our designed DFvT can not only save memory, computational cost, and accelerate the inference compared to baseline and parallel ViT, it can also achieve the best accuracy, which can attribute to the two paths design. The spatial information from SM can be efficiently integrated to CM. As a trade-off between efficiency and accuracy, decoupling design is the optimal choice.

Model	Params(M)	FLOPs(G)	FPS(imgs/s)	Memory(GB)	Top-1 Acc.(%)
a: Baseline ViT	9.8	1.4	1168.2	3.2	74.7
b: Downsampled ViT	8.8	0.5	3091.5	1.8	71.9
c: Parallel ViT	13.1	1.1	1843.3	5.0	76.9
d: DFvT-S	11.2	0.8	2203.3	2.8	78.3

Table 2. Experimental results of models with different designs corresponding to Fig. 2. Note that the ViT we use is a pyramid structure with window SA.

Spatial Module and Context Module Table 3 shows the effectiveness of each components. The model gets a moderate performance of 77.0% with only CM, which can be explained that though suffering from the lost of detailed information, the CM has extracted enough global context information for classification. After introducing the detailed low-level information from SM, the performance is improved from 77.0% to 77.5%. It is because that the SM can supply the missing local spatial information of CM, which is helpful for higher performance.

Impact of Context Module (CM) As introduced in Sec. 3.1, CM has two major components, pre-SA fusion and window-SA. Since window-SA is a necessity portion, we mainly consider the design of pre-SA.

Model	Params(M)	FLOPs(G)	Top-1 Acc.(%)	Gains(%)
CM	10.3	0.7	77.0	-
CM + SM	11.1	0.8	77.5	+0.5
CM + SM + DANE	11.2	0.8	78.3	+1.3

Table 3. Detailed performance comparison of each component in DFvT-S.

Pre-SA has two input, max-pooling features and convolutional features. Table 4 demonstrates the importance of each factor for the pre-SA fusion. The accuracy will be significantly decreased by 6.1% if removing convolutional information flow for pre-SA fusion. Similarly, we also show that the information flow from the max-pooling operation is important because the accuracy will drop by 0.2%. Obviously, convolutional features are more important for the pre-SA fusion. The reason may lie in that the convolutions provide fine-grained and diversified information, while maxpooling only retains the maximum fired neurons. Nevertheless, we still keep the max-pooling operation as it can improve the diversity of the fused information and can also contribute to the accuracy with negligible cost.

Model	Params(M)	FLOPs(G)	Top-1 Acc.(%)	Declines(%)
DFvT-S	11.2	0.8	78.3	-
w/o maxpooling info	11.2	0.8	78.1	-0.2
w/o conv info	9.6	0.5	72.2	-6.1

Table 4. Ablation study of pre-SA fusion in Context Module in terms of the number of parameters, FLOPs and top-1 accuracy on ImageNet-1K.

Impact of Spatial Module (SM) Table 3 proves that the Spatial Module can improve the performance of 0.5% with little FLOPs introduced (i.e., 0.7G to 0.8G). The depthwise convolution in SM mainly focuses on encoding fine-detailed information, thus complementing the context information from CM well.

Impact of DANE We have carefully designed the feature fusion block (i.e., DANE) for post-SA feature aggregation. To validate the designation, we compare DANE with some widely used feature fusion methods, namely, "SUM": element-wise addition, "SUM+MUL": element-wise addition and multiplication, "SE": use channel attention mechanism to fuse the features [19], "SPATIAL": use spatial attention to fuse the features [56], "rDANE": reverse version of DANE, that is, we use channel and spatial attention for inputs from CM and SM respectively. The difference is not obvious since the size of our model is not large enough. We believe that the difference will become apparent when the model size becomes larger. Note that we do not consider concatenation because it will vastly increase the parameters and computational cost. The experimental results are reported

Model	Params(M)	FLOPs(G)	Top-1 Acc.(%)	Gains(%)
SUM	11.1	0.8	77.5	-
SUM+MUL	11.1	0.8	77.6	+0.1
SE	11.2	0.8	78.0	+0.5
SPATIAL	11.2	0.8	77.8	+0.3
rDANE	11.2	0.8	78.1	+0.6
DANE	11.2	0.8	78.3	+0.8

Table 5. Different designs of the feature aggregation module to fuse the information from SM and CM.

in Table 5. The attention-based fusion strategies perform better than the pure addition or multiplication since it is not suitable to directly fuse information with different level of information representation. Among the attention based fusion operators, DANE demonstrates the best performance. This confirms our assumption that features from CM and SM exhibit different characteristics and they are complementary to each other.

5 Conclusion

In this paper, we propose a new hybrid transformer and convolution backbone, Doubly-Fused ViT (DFvT), for image classification, which retains the high accuracy of ViT but is highly efficient in computational and memory costs. The features in a standard transformer block are fast downsampled to extract context information in Context Module (CM), and is enhanced with spatial information using convolution in Spatial Module (SM). Moreover, a Dual AtteNtion Enhancement (DANE) module is used for fusion by combining spatial-wise attention for SM and channel-wise attention for CM. We also conduct model scaling to further trade off accuracy and efficiency to provide more choices for various scenarios. Experiments on ImageNet-1K image classification demonstrate that DFvT outperforms the state of the art in either accuracy or speed, or both.

6 Social Impact and Limitations

Efficient algorithms, including efficient transformer designs, are appealing for practical applications. In a world that is increasingly conscious of carbon footprint, it is particularly important to be able to reduce computational cost and its associated environmental cost, hence making AI systems more feasible for adaption. In the meantime, efficient algorithms such as efficient transformers can also be potentially used in a wider range of scenarios and bring value to a wider range of users.

Due to computational resource constraints, we have not studied DFvT design on larger scale datasets. Because the DFvT design is general and performs well under multiple settings, and transformers tend to be more data-driven, we are optimistic about its learning ability on larger scale data.

References

1. Chen, B., Li, P., Li, C., Li, B., Bai, L., Lin, C., Sun, M., Yan, J., Ouyang, W.: Glit: Neural architecture search for global and local image transformer. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 12–21 (2021)
2. Chen, C.F., Fan, Q., Panda, R.: Crossvit: Cross-attention multi-scale vision transformer for image classification. arXiv preprint arXiv:2103.14899 (2021)
3. Chen, C.F., Panda, R., Fan, Q.: Regionvit: Regional-to-local attention for vision transformers. arXiv preprint arXiv:2106.02689 (2021)
4. Chen, P., Chen, Y., Liu, S., Yang, M., Jia, J.: Exploring and improving mobile level vision transformers. arXiv preprint arXiv:2108.13015 (2021)
5. Chen, Y., Dai, X., Chen, D., Liu, M., Dong, X., Yuan, L., Liu, Z.: Mobile-former: Bridging mobilenet and transformer. arXiv preprint arXiv:2108.05895 (2021)
6. Chen, Z., Xie, L., Niu, J., Liu, X., Wei, L., Tian, Q.: Visformer: The vision-friendly transformer. arXiv preprint arXiv:2104.12533 (2021)
7. Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., Shen, C.: Twins: Revisiting spatial attention design in vision transformers. arXiv preprint arXiv:2104.13840 (2021)
8. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 764–773 (2017)
9. d'Ascoli, S., Touvron, H., Leavitt, M., Morcos, A., Biroli, G., Sagun, L.: Convit: Improving vision transformers with soft convolutional inductive biases. arXiv preprint arXiv:2103.10697 (2021)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
12. El-Nouby, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., et al.: Xcit: Cross-covariance image transformers. arXiv preprint arXiv:2106.09681 (2021)
13. Gulati, A., Qin, J., Chiu, C.C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., et al.: Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100 (2020)
14. Guo, J., Han, K., Wu, H., Xu, C., Tang, Y., Xu, C., Wang, Y.: Cmt: Convolutional neural networks meet vision transformers. arXiv preprint arXiv:2107.06263 (2021)
15. Han, K., Guo, J., Tang, Y., Wang, Y.: Pyramidtnt: Improved transformer-in-transformer baselines with pyramid architecture (2022)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
17. Heo, B., Yun, S., Han, D., Chun, S., Choe, J., Oh, S.J.: Rethinking spatial dimensions of vision transformers. arXiv preprint arXiv:2103.16302 (2021)
18. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)

19. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7132–7141 (2018)
20. Jiang, Z., Hou, Q., Yuan, L., Zhou, D., Jin, X., Wang, A., Feng, J.: Token labeling: Training a 85.5% top-1 accuracy vision transformer with 56m parameters on imagenet. arXiv preprint arXiv:2104.10858 (2021)
21. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
22. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. vol. 25, pp. 1097–1105 (2012)
23. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942 (2019)
24. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998). https://doi.org/10.1109/5.726791
25. Li, K., Wang, Y., Zhang, J., Gao, P., Song, G., Liu, Y., Li, H., Qiao, Y.: Uniformer: Unifying convolution and self-attention for visual recognition. arXiv preprint arXiv:2201.09450 (2022)
26. Li, Y., Zhang, K., Cao, J., Timofte, R., Van Gool, L.: Localvit: Bringing locality to vision transformers. arXiv preprint arXiv:2104.05707 (2021)
27. Li, Y., Yao, T., Pan, Y., Mei, T.: Contextual transformer networks for visual recognition. arXiv preprint arXiv:2107.12292 (2021)
28. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021)
29. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s (2022)
30. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3431–3440 (2015)
31. Ma, N., Zhang, X., Zheng, H.T., Sun, J.: Shufflenet v2: Practical guidelines for efficient cnn architecture design. In: Proceedings of the European Conference on Computer Vision. pp. 116–131 (2018)
32. Mehta, S., Rastegari, M.: Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. arXiv preprint arXiv:2110.02178 (2021)
33. Nie, D., Xue, J., Ren, X.: Bidirectional pyramid networks for semantic segmentation. In: Proceedings of the Asian Conference on Computer Vision (2020)
34. Pan, Z., Zhuang, B., Liu, J., He, H., Cai, J.: Scalable vision transformers with hierarchical pooling. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 377–386 (2021)
35. Peng, C., Zhang, X., Yu, G., Luo, G., Sun, J.: Large kernel matters–improve semantic segmentation by global convolutional network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4353–4361 (2017)
36. Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P.: Designing network design spaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10428–10436 (2020)
37. Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., Dosovitskiy, A.: Do vision transformers see like convolutional neural networks? vol. 34 (2021)

38. Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., Hsieh, C.J.: Dynamicvit: Efficient vision transformers with dynamic token sparsification. arXiv preprint arXiv:2106.02034 (2021)
39. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. vol. 28, pp. 91–99 (2015)
40. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 234–241 (2015)
41. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4510–4520 (2018)
42. Santurkar, S., Tsipras, D., Ilyas, A., Madry, A.: How does batch normalization help optimization? In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. pp. 2488–2498 (2018)
43. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
44. Srinivas, A., Lin, T.Y., Parmar, N., Shlens, J., Abbeel, P., Vaswani, A.: Bottleneck transformers for visual recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 16519–16529 (2021)
45. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015)
46. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: Proceedings of the International Conference on Machine Learning. pp. 6105–6114 (2019)
47. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: Proceedings of the International Conference on Machine Learning. pp. 10347–10357 (2021)
48. Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., Jégou, H.: Going deeper with image transformers. arXiv preprint arXiv:2103.17239 (2021)
49. Vaswani, A., Ramachandran, P., Srinivas, A., Parmar, N., Hechtman, B., Shlens, J.: Scaling local self-attention for parameter efficient visual backbones. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12894–12904 (2021)
50. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Proceedings of the Advances in Neural Information Processing Systems. pp. 5998–6008 (2017)
51. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., Xiao, B.: Deep high-resolution representation learning for visual recognition. TPAMI (2019)
52. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pvtv2: Improved baselines with pyramid vision transformer. arXiv preprint arXiv:2106.13797 (2021)
53. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. arXiv preprint arXiv:2102.12122 (2021)
54. Wang, W., Yao, L., Chen, L., Cai, D., He, X., Liu, W.: Crossformer: A versatile vision transformer based on cross-scale attention. arXiv preprint arXiv:2108.00154 (2021)

55. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7794–7803 (2018)
56. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision. pp. 3–19 (2018)
57. Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., Zhang, L.: Cvt: Introducing convolutions to vision transformers. arXiv preprint arXiv:2103.15808 (2021)
58. Wu, K., Peng, H., Chen, M., Fu, J., Chao, H.: Rethinking and improving relative position encoding for vision transformer. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 10033–10041 (2021)
59. Xiao, T., Singh, M., Mintun, E., Darrell, T., Dollár, P., Girshick, R.: Early convolutions help transformers see better. arXiv preprint arXiv:2106.14881 (2021)
60. Yan, H., Li, Z., Li, W., Wang, C., Wu, M., Zhang, C.: Contnet: Why not use convolution and transformer at the same time? arXiv preprint arXiv:2104.13497 (2021)
61. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: Xlnet: Generalized autoregressive pretraining for language understanding. vol. 32 (2019)
62. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European Conference on Computer Vision (2018)
63. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122 (2015)
64. Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., Yan, S.: Metaformer is actually what you need for vision. arXiv preprint arXiv:2111.11418 (2021)
65. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.H., Tay, F.E., Feng, J., Yan, S.: Tokens-to-token vit: Training vision transformers from scratch on imagenet. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 558–567 (2021)
66. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6023–6032 (2019)
67. Zhang, P., Dai, X., Yang, J., Xiao, B., Yuan, L., Zhang, L., Gao, J.: Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. arXiv preprint arXiv:2103.15358 (2021)
68. Zhang, Q., Yang, Y.: Rest: An efficient transformer for visual recognition. arXiv preprint arXiv:2105.13677v3 (2021)
69. Zhang, X., Zhou, X., Lin, M., Sun, J.: Shufflenet: An extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6848–6856 (2018)