MixFormer: Mixing Features across Windows and Dimensions

Qiang Chen[1*], Qiman Wu[1*], Jian Wang[1*], Qinghao Hu[2†], Tao Hu[1]
Errui Ding[1], Jian Cheng[2], Jingdong Wang[1]
[1]Baidu VIS
[2]NLPR, Institute of Automation, Chinese Academy of Sciences
{chenqiang13,wuqiman,wangjian33,hutao06,dingerrui,wangjingdong}@baidu.com
huqinghao2014@ia.ac.cn, jcheng@nlpr.ia.ac.cn

Models	#Channels	#Blocks	#Heads
MixFormer-B0	$C = 24$	$[1, 2, 6, 6]$	$[3, 6, 12, 24]$
MixFormer-B1	$C = 32$	$[1, 2, 6, 6]$	$[2, 4, 8, 16]$
MixFormer-B2	$C = 32$	$[2, 2, 8, 8]$	$[2, 4, 8, 16]$
MixFormer-B3	$C = 48$	$[2, 2, 8, 6]$	$[3, 6, 12, 24]$
MixFormer-B4	$C = 64$	$[2, 2, 8, 8]$	$[4, 8, 16, 32]$
MixFormer-B5	$C = 96$	$[1, 2, 8, 6]$	$[6, 12, 24, 48]$
MixFormer-B6	$C = 96$	$[2, 4, 16, 12]$	$[6, 12, 24, 48]$

Table 1. **Architecture Variants.** Detailed configurations of architecture variants of MixFormer.

Abstract

In this supplementary file, we provide limitations of MixFormer, more model variants, experimental details, and discussions with related works.

A. Limitations

Our MixFormer is proposed to mitigate the issues in local-window self-attention [15, 25]. Thus it may be limited to window-based vision transformers in this paper. Although the parallel design and the bi-directional interactions can be applied to the global self-attention [5, 24], it is not clear that how many gains can the above designs bring. We conduct a simple experiment on DeiT-Tiny [24]. But the result becomes slightly worse, as shown in Table 4. More efforts are needed to apply our mixing block to global attention. We leave this for future work. Moreover, we build the MixFormer series manually, limiting MixFormer in existing instances. Other methods such as NAS (Network Architecture Search) [23] can be applied to further improve the results.

B. More Variants of MixFormer

We scale our MixFormer to smaller and larger models. In this section, we provide two instantiated models (MixFormer-B0 and MixFormer-B5). Their detailed settings are provided in Table 1, along with previous methods (from B1 to B4). Note that MixFormer-B0 and MixFormer-B5 are two examples. More variants can be obtained with further attempts following the design of MixFormer. Then, we validate their effectiveness on ImageNet-1K [4]. The results are illustrated in Table 2.

On one side, MixFormer-B0 achieves competitive result (76.5% Top-1 accuracy on ImageNet-1K [4]) even with 0.4G FLOPs, which lies in the mobile level [18, 22]. While other vision transformer variants [15, 26–29] did not provide a range of model sizes like our MixFormer, especially in mobile level. We believe that further efforts can be made to give higher performance to achieve state-of-the-art results [10, 23] in mobile level models. On the other side, MixFormer-B5 shows an example to scale our MixFormer to larger models. It has 6.8G FLOPs, while it can achieve on par results with Swin-B (15.4G) [15], Focal-S (9.1G) [29], Shuffle-S (8.9G) [12], and EfficientNet-B5 (9.9G) [23], which demonstrates the computational efficiency of MixFormer. MixFormer-B6 achieves **83.8%** top-1 accuracy on ImageNet-1K [4]. It maintains the superior performance to Swin-B(15.4G) [15] and is comparable to other models with less flops.

The above results verify the scalability of MixFormer to smaller and larger models. Moreover, it has the potential for further improvements.

C. Additional Experiments

Window Sizes in Local-Window Self-Attention. We conduct ablation study on the window size in local-window self-attention with MixFormer-B1. The experimental settings are follow the ones in the ablation studies. The results in Table 3 show that larger window size (ws=12) achieves on par performance with ws=7 (78.4%) on ImageNet-1K [4]. Based on the above result, We follow the con- ventional design of Swin Transformer (ws=7) [15] in all vari-

*Equal Contribution.
†Corresponding author.

Method	#Params	FLOPs	Top-1
ConvNets			
RegNetY-4G [21]	21M	4.0G	80.0
RegNetY-8G [21]	39M	8.0G	81.7
RegNetY-16G [21]	84M	16.0G	82.9
EffNet-B0 [23]	5M	0.4G	77.1
EffNet-B1 [23]	8M	0.7G	79.1
EffNet-B2 [23]	9M	1.0G	80.1
EffNet-B3 [23]	12M	1.8G	81.6
EffNet-B4 [23]	19M	4.2G	82.9
EffNet-B5 [23]	30M	9.9G	83.6
Vision Transformers			
DeiT-T [24]	6M	1.3G	72.2
DeiT-S [24]	22M	4.6G	79.9
DeiT-B [24]	87M	17.5G	81.8
PVT-T [27]	13M	1.8G	75.1
PVT-S [27]	25M	3.8G	79.8
PVT-M [27]	44M	6.7G	81.2
PVT-L [27]	61M	9.8G	81.7
CvT-13 [28]	20M	4.5G	81.6
CvT-21 [28]	32M	7.1G	82.5
TwinsP-S [2]	24M	3.8G	81.2
DS-Net-S [19]	23M	3.5G	82.3
Swin-T [15]	29M	4.5G	81.3
Swin-S [15]	50M	8.7G	83.0
Swin-B [15]	88M	15.4G	83.5
Twins-S [2]	24M	2.9G	81.7
Twins-B [2]	56M	8.6G	83.2
LG-T [13]	33M	4.8G	82.1
LG-S [13]	61M	9.4G	83.3
Focal-T [29]	29M	4.9G	82.2
Focal-S [29]	51M	9.1G	83.5
Shuffle-T [12]	29M	4.6G	82.5
Shuffle-S [12]	50M	8.9G	83.5
MixFormer-B0 (**Ours**)	5M	0.4G	76.5
MixFormer-B1 (**Ours**)	8M	0.7G	78.9
MixFormer-B2 (**Ours**)	10M	0.9G	80.0
MixFormer-B3 (**Ours**)	17M	1.9G	81.7
MixFormer-B4 (**Ours**)	35M	3.6G	83.0
MixFormer-B5 (**Ours**)	62M	6.8G	83.5
MixFormer-B6 (**Ours**)	119M	12.7G	83.8

Table 2. **Classification accuracy on the ImageNet validation set.** Performances are measured with a single 224×224 crop. "Params" refers to the number of parameters. "FLOPs" is calculated under the input scale of 224×224.

Window Sizes	ImageNet	
	Top-1	Top-5
7×7	78.4	94.3
$\mathbf{12 \times 12}$	78.4	94.5

Table 3. **Window Sizes in Local-window Self-attention.** We investigate various window sizes for Local-window Self-attention in MixFormer.

ants of MixFormer.

Apply Mixing Block to DeiT. Although our mixing block is proposed to solve the window connection problem in local-window self-attention [15]. It can also be applied to global attentions [5, 24]. We simply apply our mixing block

DeiT-Tiny [24]	ImageNet	
	Top-1	Top-5
Baseline	72.2	91.1
Baseline+Mixing Block	71.3	90.5

Table 4. **Apply Mixing Block to DeiT-Tiny.** We apply our mixing block to global attention.

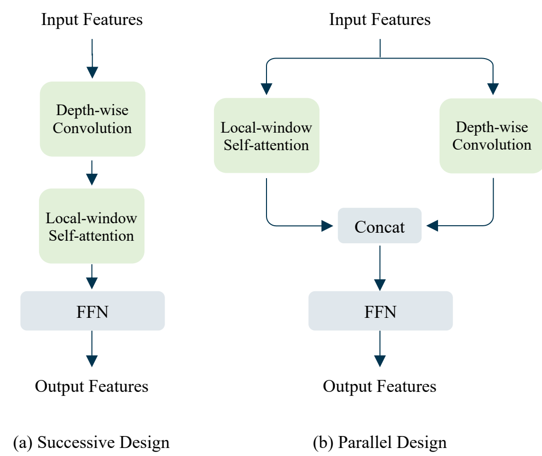

Figure 1. **Successive Design and Parallel Design.** We combine local-window self-attention with depth-wise convolution in two different ways. Other details in the block, such as module design, normalization layers, and shortcuts, are omitted for a neat presentation.

to Deit-Tiny [24]. But the result is slightly lower than baseline (71.3% vs. 72.2%) on ImageNet-1K [4]. We conjecture that global attention (ViT-based model) may not share the same problem and detailed design for global attention may need further investigating. We leave this for future work.

D. Detailed Experimental Settings

Successive Design and Parallel Design. In Figure 1, we give the details on how to combine local-window self-attention and depth-wise convolution in the successive design and the parallel design. To make a fair comparison, we adjust the channels in the blocks to keep the computational complexity the same in the two designs.

Image Classification. We train all models for 300 epochs with an image size of 224×224 on ImageNet-1K [4]. We adjust the training settings gently when training models in different sizes. The detailed setting is in Table 5.

Object Detection and Instance Segmentation. When transferring MixFormer to object detection and instance segmentation on MS COCO [14], we consider two typical frameworks: Mask R-CNN [9] and Cascade Mask R-CNN [1, 9]. We adopt AdamW [17] optimizer with an initial learning rate of 0.0002 and a batch size of 16. To make

config	value
optimizer	AdamW [17]
base learning rate	8e-4 (B0-B3), 1e-3 (B4, B5, B6)
weight decay	0.04 (B0-B3), 0.05 (B4, B5, B6)
optimizer momentum	β_1, β_2=$0.9, 0.999$
batch size	1024
learning rate schedule	cosine decay [16]
minimum learning rate	1e-6
warmup epochs	20 (B0-B4), 40 (B5, B6)
warmup learning rate	1e-7
training epochs	300
augmentation	RandAug(9, 0.5) [3]
color jitter	0.4
mixup [32]	0.2
cutmix [31]	1.0
random erasing [33]	0.25
drop path [11]	[0.0, 0.05, 0.1, 0.2, 0.3, 0.5] (B0-B6)

Table 5. **Image Classification Training Settings.**

config	value
optimizer	AdamW
base learning rate	0.0002
weight decay	0.04 (B0-B3), 0.05 (B4, B5)
optimizer momentum	β_1, β_2=$0.9, 0.999$
batch size	16
learning rate schedule	steps:[8, 11] (1×), [27, 33] (3×)
warmup iterations (ratio)	500 (0.001)
training epochs	12 (1×), 36 (3×)
scales	(800, 1333) (1×), Multi-scales [15] (3×)
drop path	0.0 (B0-B3), 0.1 (B4, B5)

Table 6. **Object Detection and Instance Segmentation Training Settings.**

a fair comparison with other works, we make all normalization layers trainable in MixFormer[1]. When training different sizes of models, we adjust the training settings gently according to their settings used in image classification. Table 6 shows the detailed hyper-parameters used in training models on MS COCO [14].

Semantic Segmentation. On ADE20K [34], we use the AdamW optimizer [17] with an initial learning rate 0.00006, a weight decay 0.01, and a batch size of 16. We train all models for 160K on ADE20K. For testing, we report the results with single-scale testing and multi-scale testing on main comparisons, while we only give single-scale testing results on ablation studies. In multi-scale testing, the resolutions used are the $[0.5, 0.75, 1.0, 1.25, 1.5, 1.75]\times$ of that in training. The settings mainly follow [15]. For the path drop rates in different models, we adopt the same hyper-parameters as in MS COCO [14].

Keypoint Detection. We conduct experiments on the MS COCO human pose estimation benchmark. We train the models for 210 epochs with an AdamW optimizer, an image size of 256×192, and a batch size of 256. The training and evaluation hyper-parameters are mostly following the ones in HRFormer [30].

Long-tail Instance Segmentation. We use the hyper-parameters of Mask R-CNN [9] on MS COCO [14] when training models for long-tail instance segmentation on LVIS [7]. We report the results with a $1\times$ schedule. The training augmentations and sampling methods are the same for all models, which adopt a multi-scale training and use balanced sampling by following [7].

[1] Wherever BN is applied, we use *synchronous* BN across all GPUs.

E. Discussion with Related Works

In MixFormer, we consider two types of information exchanges: (1) across dimensions, (2) across windows.

For the first type, Conformer [20] also performs information exchange between a transformer branch and a convolution branch. While its motivation is different from ours. Conformer aims to couple local and global features across convolution and transformer branches. MixFormer uses channel and spatial interactions to address the weak modeling ability issues caused by weight sharing on the channel (local-window self-attention) and the spatial (depth-wise convolution) dimensions [8].

For the second type, Twins (strided convolution + global sub-sampled attention) [2] and Shuffle Transformer (neighbor-window connection (NWC) + random spatial shuffle) [12] construct local and global connections to achieve information exchanges, MSG Transformer (channel shuffle on extra MSG tokens) [6] applies global connection. Our MixFormer achieves this goal by concatenating the parallel features: the non-overlapped window feature and the local-connected feature (output of the dwconv3x3).

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

[2] 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. *arXiv preprint arXiv:2104.13840*, 1(2):3, 2021. 2, 3

[3] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 3

[4] 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. 1, 2

[5] 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. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2

[6] Jiemin Fang, Lingxi Xie, Xinggang Wang, Xiaopeng Zhang, Wenyu Liu, and Qi Tian. Msg-transformer: Exchanging local spatial information by manipulating messenger tokens. *arXiv preprint arXiv:2105.15168*, 2021. 3

[7] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5356–5364, 2019. 3

[8] Qi Han, Zejia Fan, Qi Dai, Lei Sun, Ming-Ming Cheng, Jiaying Liu, and Jingdong Wang. Demystifying local vision transformer: Sparse connectivity, weight sharing, and dynamic weight. *arXiv preprint arXiv:2106.04263*, 2021. 3

[9] 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. 2, 3

[10] 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. 1

[11] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European conference on computer vision*, pages 646–661. Springer, 2016. 3

[12] 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. 1, 2, 3

[13] Jinpeng Li, Yichao Yan, Shengcai Liao, Xiaokang Yang, and Ling Shao. Local-to-global self-attention in vision transformers. *arXiv preprint arXiv:2107.04735*, 2021. 2

[14] 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. 2, 3

[15] 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. *arXiv preprint arXiv:2103.14030*, 2021. 1, 2, 3

[16] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 3

[17] I. Loshchilov and F. Hutter. Fixing weight decay regularization in adam. 2017. 2, 3

[18] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018. 1

[19] Mingyuan Mao, Renrui Zhang, Honghui Zheng, Peng Gao, Teli Ma, Yan Peng, Errui Ding, and Shumin Han. Dual-stream network for visual recognition. *arXiv preprint arXiv:2105.14734*, 2021. 2

[20] Zhiliang Peng, Wei Huang, Shanzhi Gu, Lingxi Xie, Yaowei Wang, Jianbin Jiao, and Qixiang Ye. Conformer: Local features coupling global representations for visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 367–376, 2021. 3

[21] 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. 2

[22] 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. 1

[23] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019. 1, 2

[24] 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. 1, 2

[25] Ashish Vaswani, Prajit Ramachandran, Aravind Srinivas, Niki Parmar, Blake Hechtman, and Jonathon Shlens. Scaling local self-attention for parameter efficient visual backbones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12894–12904, 2021. 1

[26] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvtv2: Improved baselines with pyramid vision transformer. *arXiv preprint arXiv:2106.13797*, 2021. 1

[27] 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. *arXiv preprint arXiv:2102.12122*, 2021. 1, 2

[28] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021. 1, 2

[29] 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. *arXiv preprint arXiv:2107.00641*, 2021. 1, 2

[30] Yuhui Yuan, Rao Fu, Lang Huang, Weihong Lin, Chao Zhang, Xilin Chen, and Jingdong Wang. Hrformer: High-resolution transformer for dense prediction. *Advances in Neural Information Processing Systems*, 2021. 3

[31] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 3

[32] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 3

[33] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings*

of the AAAI Conference on Artificial Intelligence, number 07, pages 13001–13008, 2020. 3

[34] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 3