Mobile-Former: Bridging MobileNet and Transformer

Yinpeng Chen[1] Xiyang Dai[1] Dongdong Chen[1] Mengchen Liu[1] Xiaoyi Dong[2]
Lu Yuan[1] Zicheng Liu[1]

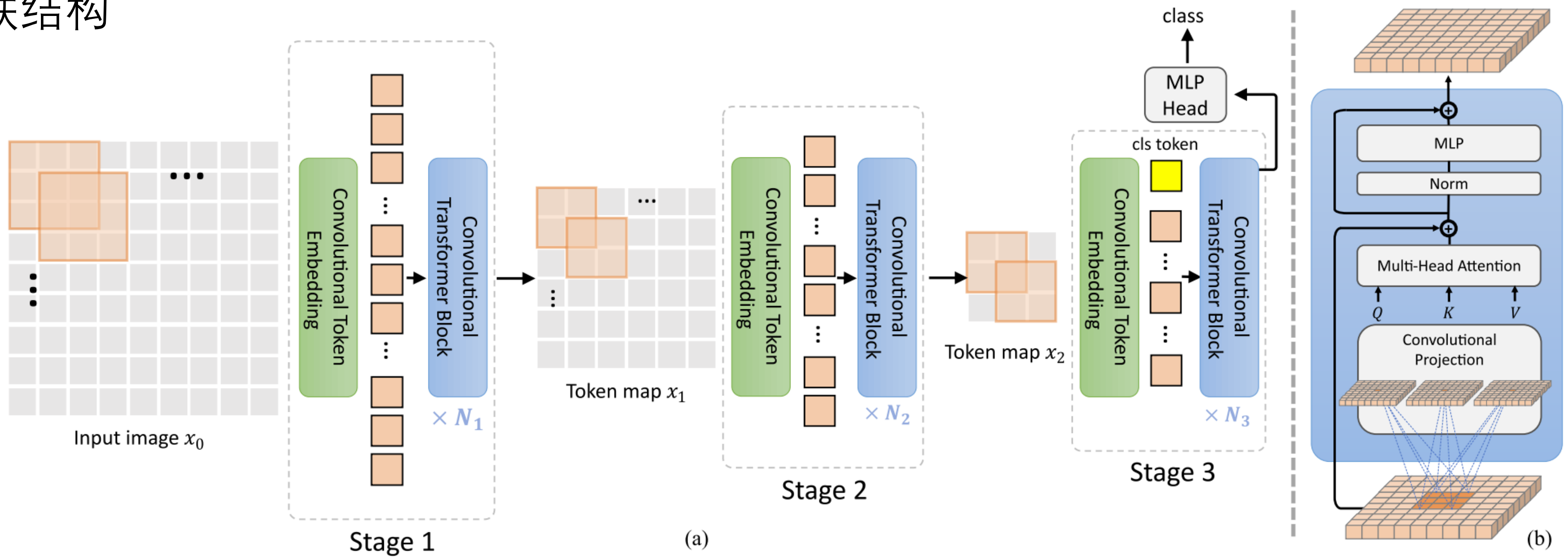
[1] Microsoft [2] University of Science and Technology of China
{yiche,xidai,dochen,mengcliu,luyuan,zliu}@microsoft.com, dlight@mail.ustc.edu.cn

CVPR 2022 Oral

Y. Chen et al., "Mobile-Former: Bridging MobileNet and Transformer," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 5260-5269, doi: 10.1109/CVPR52688.2022.00520.

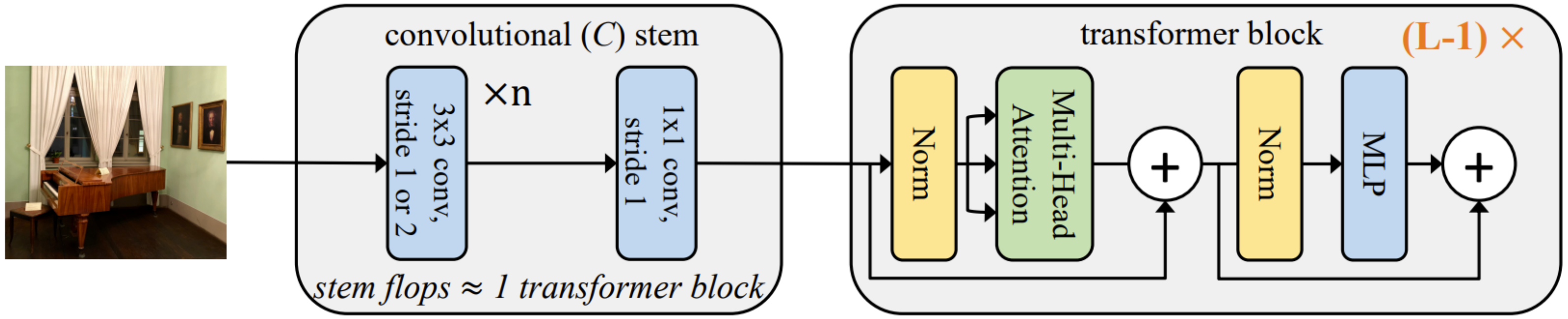
How to design **efficient** networks to **effectively** encode both local precessing and global interaction?

文章背景：串联结构



！ Transformer带来的浮点运算量仍然很大

intertwining convolution into each transformer block



！不能及时的将提取到的局部特征和全局信息进行融合

Use convolution at the beginning and then use visual transformer

[1] Wu H, Xiao B, Codella N, et al. Cvt: Introducing convolutions to vision transformers[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 22-31.
[2] Xiao T, Singh M, Mintun E, et al. Early convolutions help transformers see better[J]. Advances in Neural Information Processing Systems, 2021, 34: 30392-30400.

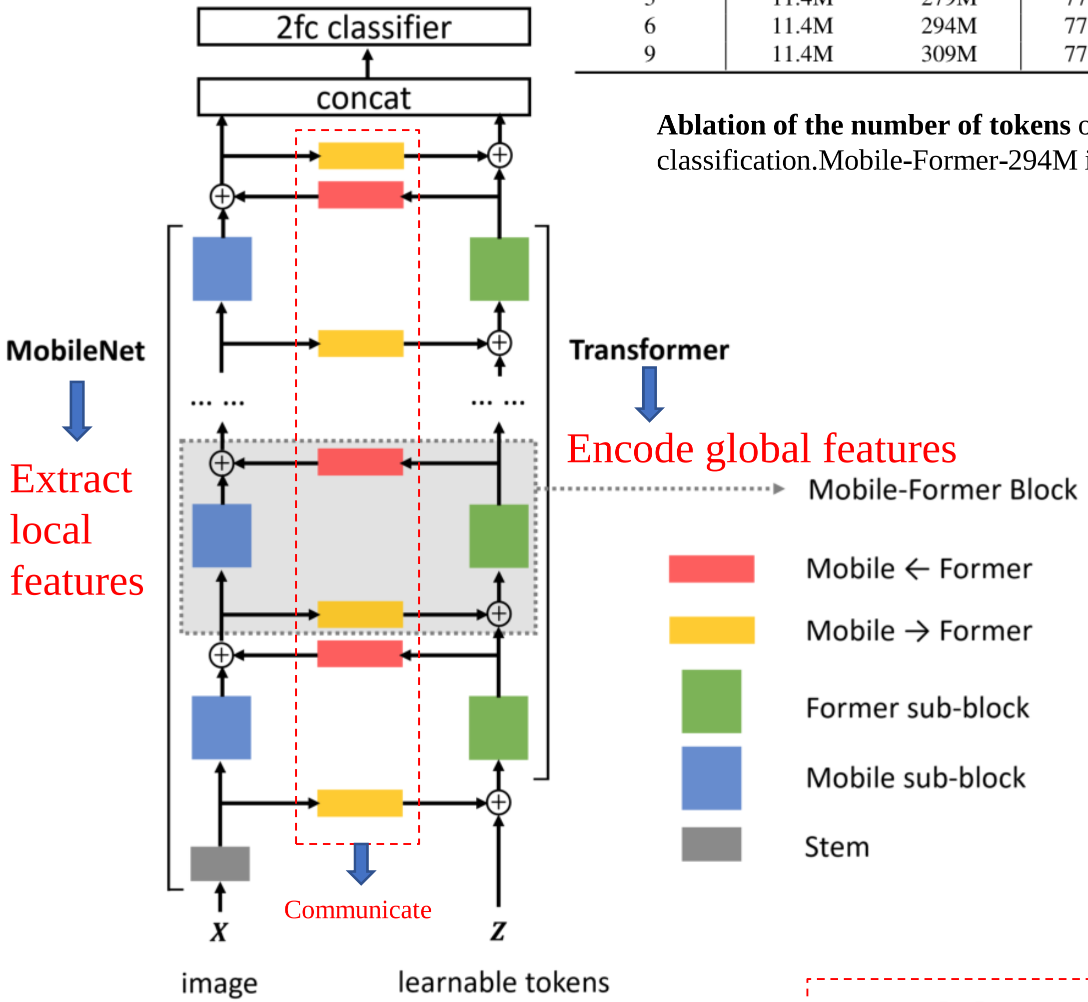
How to design **efficient** networks to **effectively** encode both local precessing and global interaction?

Mobile-former: series to parallel(*)

Advantages:

- 并行结构可以同时利用MobileNet和Transformer的优势来对局部特征和全局特征进行处理，提高表征能力
- 轻量化交叉注意力来建模双向桥来融合局部和全局信息，Mobile-former不仅计算高效，并且具有更加强大的表征能力。

The bridge and Former consume less than 20% of the total computational cost,but significantly improve the representation capability.



#Tokens	#Params	MAdds	Top-1	Top-5
1	11.4M	269M	77.1	93.2
3	11.4M	279M	77.6	93.6
6	11.4M	294M	77.8	93.7
9	11.4M	309M	77.7	93.8

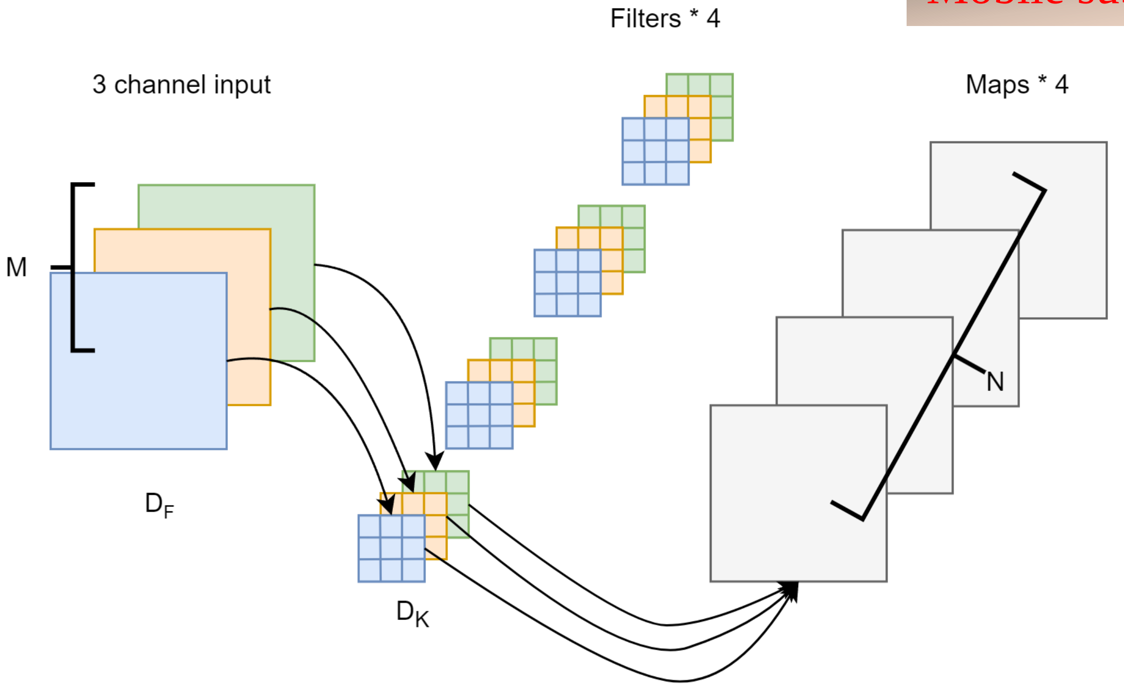
Ablation of the number of tokens on ImageNet classification.Mobile-Former-294M is used.

$X \in \mathbb{R}^{HW \times 3}$ $Z \in \mathbb{R}^{M \times d}$ $(M \leq 6)$

M and d are identival across all blocks

（$Z \in$ 6×192 in 294M FLOPs）

Mobile sub-block (背景)



$$\frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F}{D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F}$$

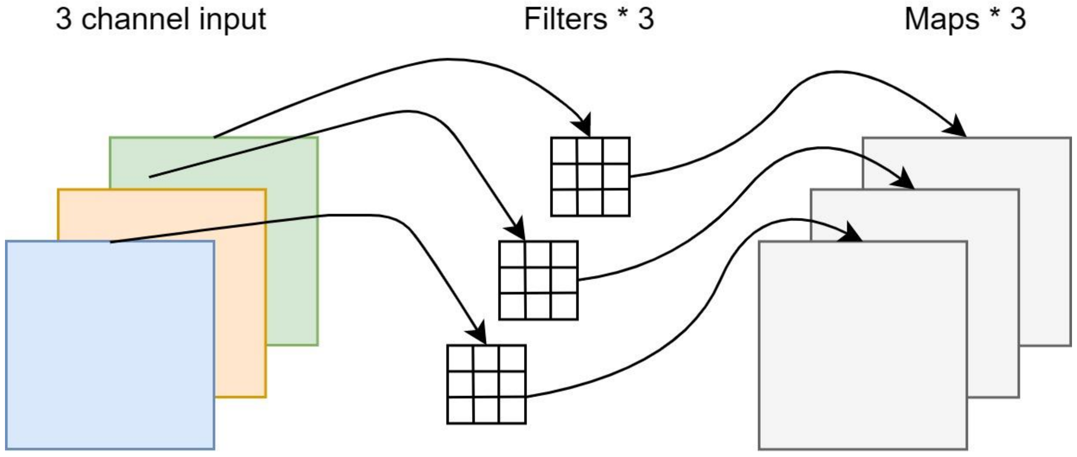
DW+PW

普通卷积

默认步距为1

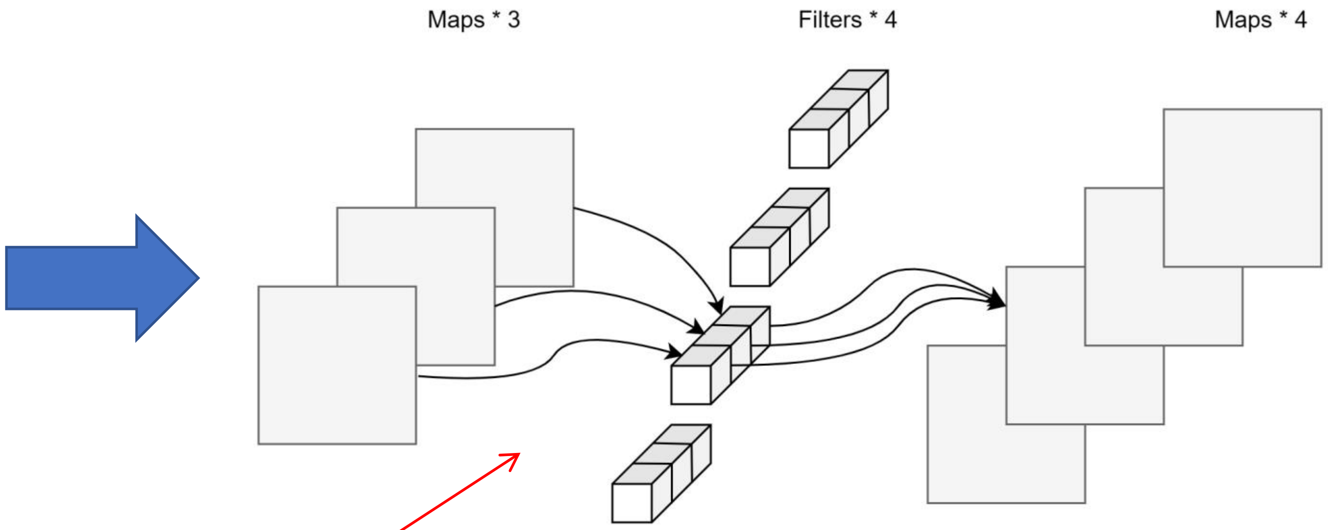
$$= \frac{1}{N} + \frac{1}{D_K^2} = \frac{1}{N} + \frac{1}{9}$$

理论上普通卷积计算量是DW+PW的8到9倍



DW卷积
Depthwise Conv

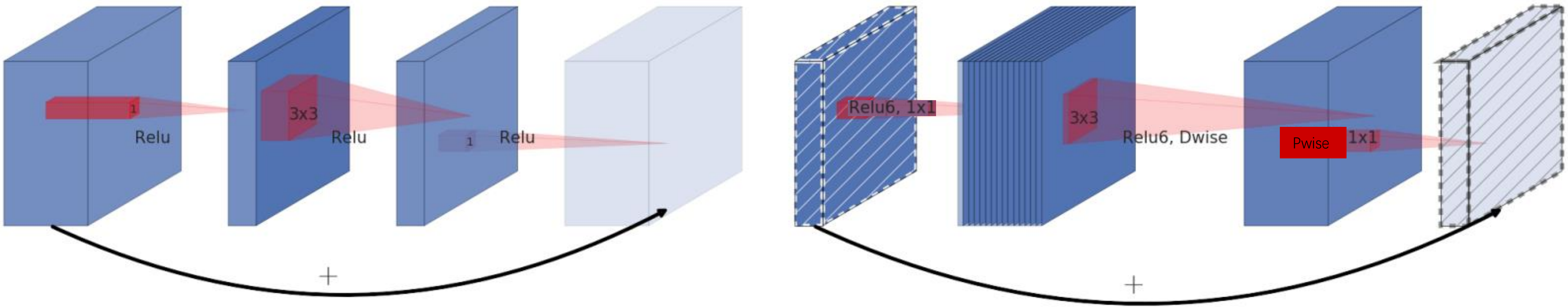
一个卷积核仅负责一个通道的特征提取



融合不同通道中提取到的特征

PW卷积
Pointwise Conv

Mobile sub-block (背景)

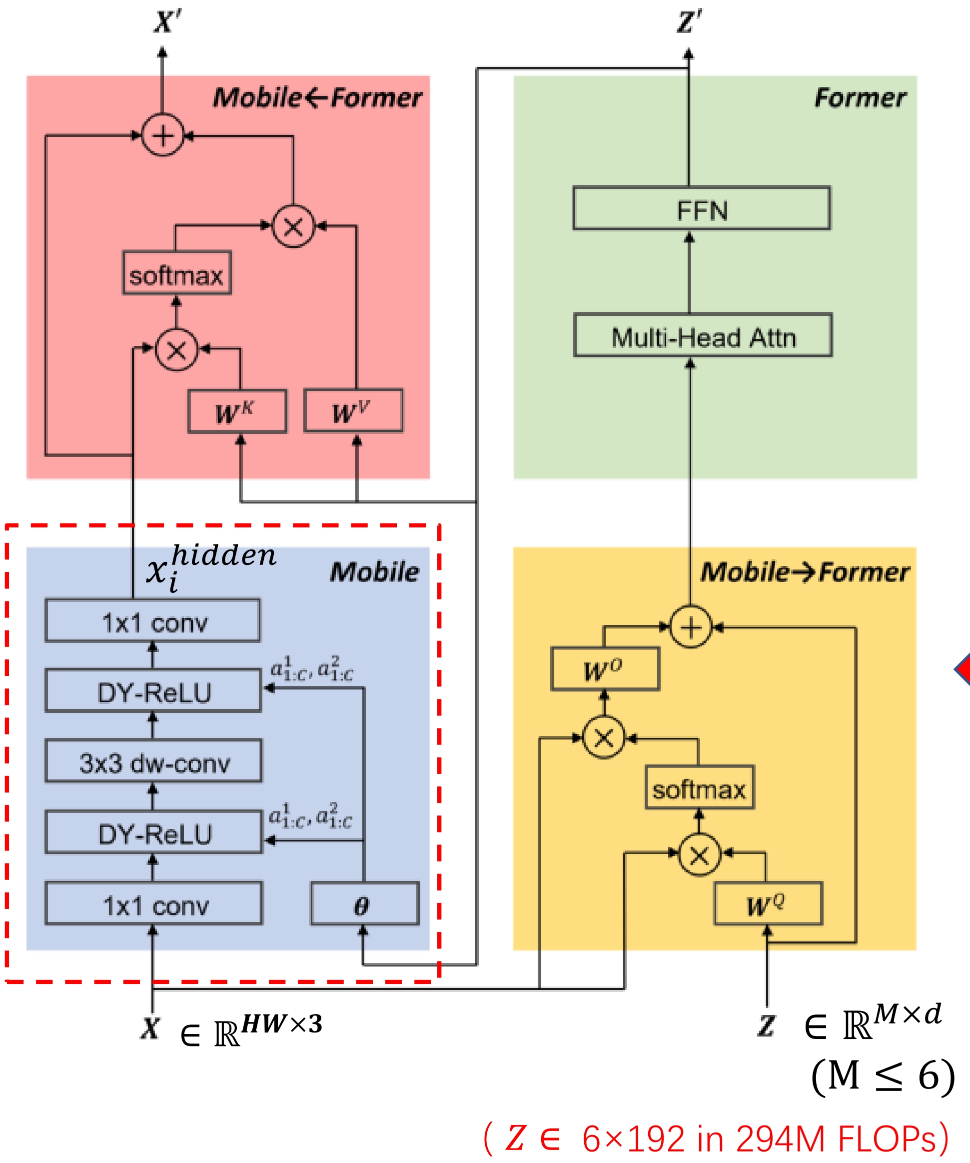


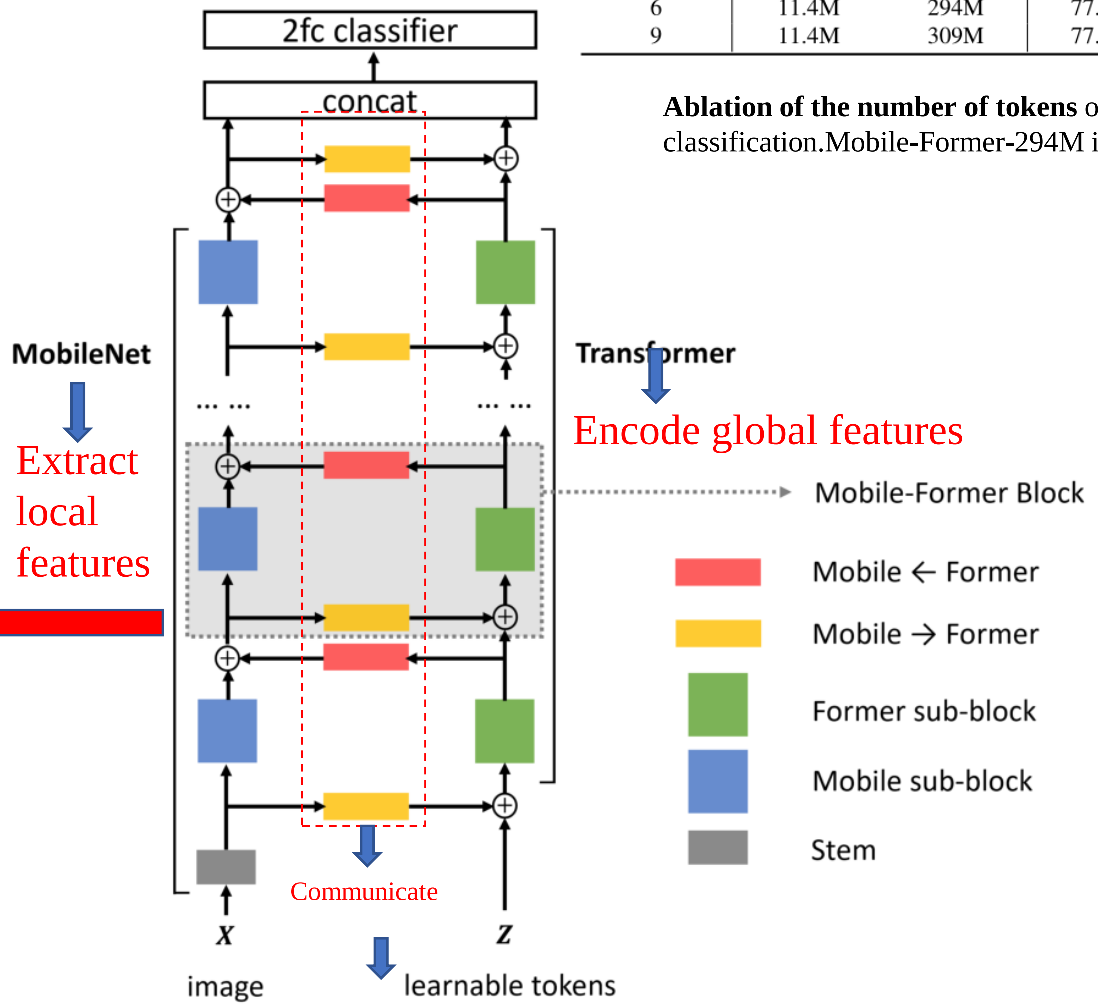
Residual block

Inverted residual block

- 倒残差结构提升dw卷积的维度，进而提高特征的表达能力
- 当信息从高维空间经过非线性映射到低维空间时，会发生信息坍塌(丢失)，所以在倒残差结构中，进行降维操作时，使用线性激活函数

Mobile sub-block



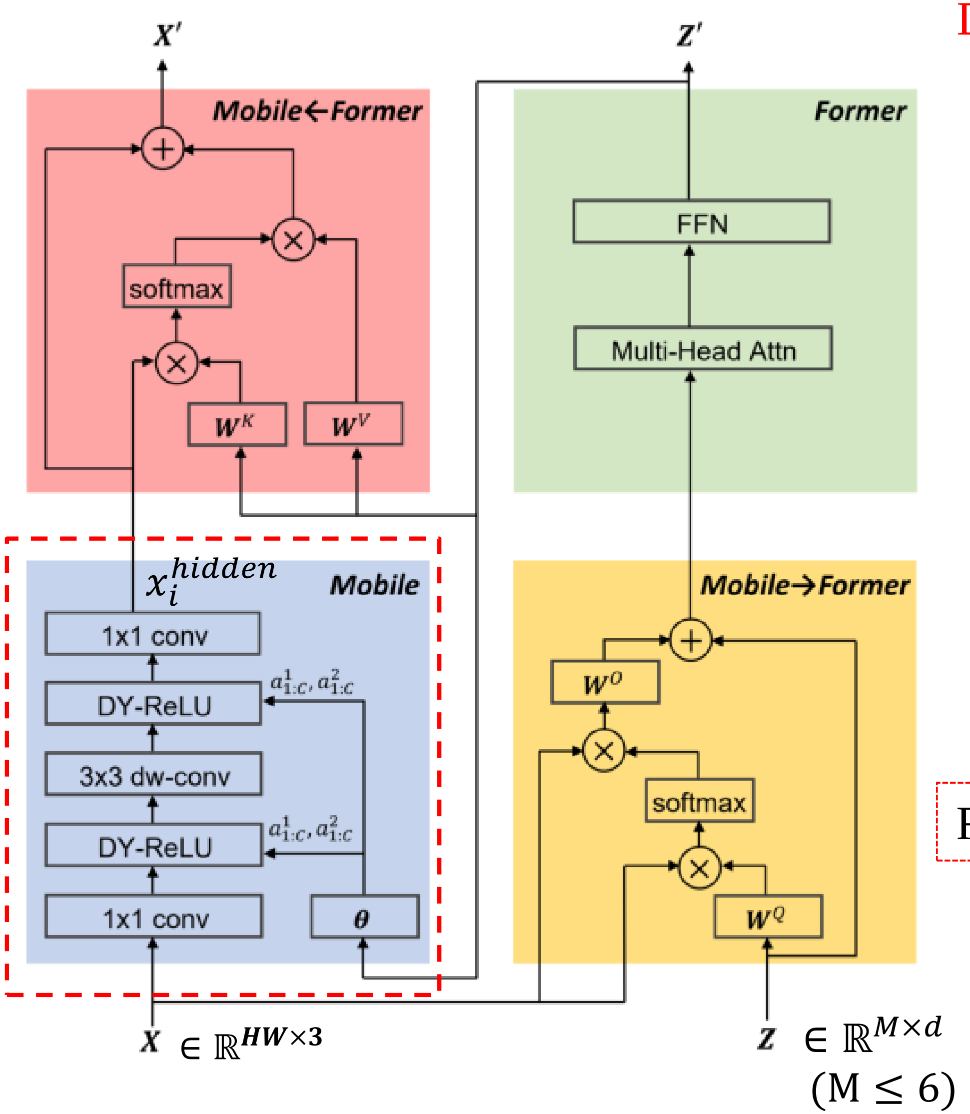


#Tokens	#Params	MAdds	Top-1	Top-5
1	11.4M	269M	77.1	93.2
3	11.4M	279M	77.6	93.6
6	11.4M	294M	77.8	93.7
9	11.4M	309M	77.7	93.8

Ablation of the number of tokens on ImageNet classification.Mobile-Former-294M is used.

Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z. (2020). Dynamic ReLU. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science(), vol 12364. Springer, Cham.

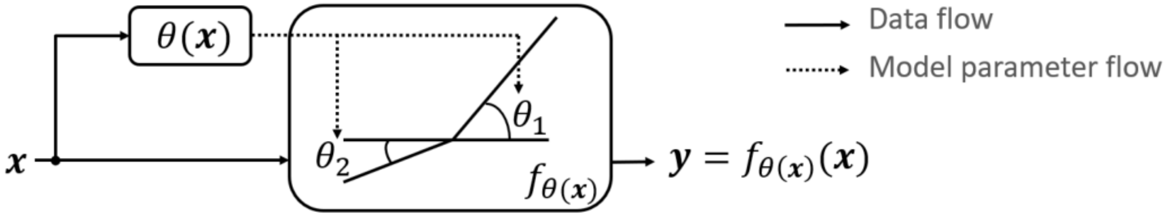
Mobile sub-block



（$Z \in$ 6×192 in 294M FLOPs）

Dynamic-relu:

Definition of Dynamic ReLU:



$$y_c = f_{\theta(x)}(x_c) = \max_{1\le k\le K}\{a_c^k(x)x_c + b_c^k(x)\}$$ ⇦ 带参分段线性函数

（In Mobile sub-block: K = 2）

$$y_c = \max\{x_c, 0\} = max_k\{a_c^k x_c + b_c^k\}(k = 2, a_c^1 = 1, b_c^1 = 0, a_c^2 = 0, b_c^2 = 0)$$

⇧

Relu

the coefficients (a_c^k, b_c^k) are the output of a hyper function $\theta(x)$ as:

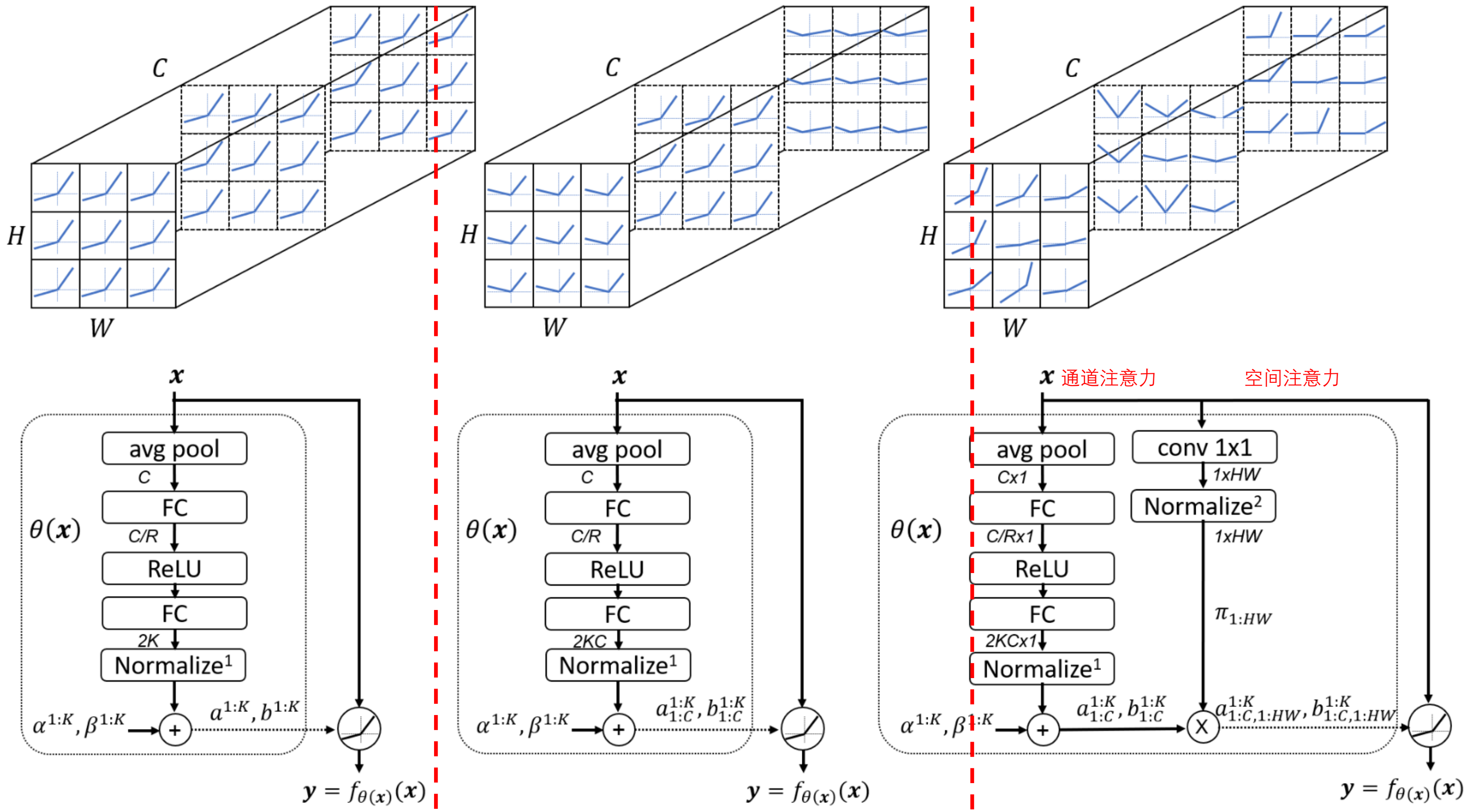
$$[a_1^1, \ldots, a_C^1, \ldots, a_1^K, \ldots, a_C^K, b_1^1, \ldots, b_C^1, \ldots, b_1^K, \ldots, b_C^K]^T = \theta(x)$$

Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z. (2020). Dynamic ReLU. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science(), vol 12364. Springer, Cham.

Mobile sub-block

超函数θ内部计算方式(3种):

(将所有输入元素x={x_c}的全局上下文编码在超函数$\theta(x)$中，以适应激活函数$f_{\theta(x)}(x_c)$)



a) **DY-ReLU-A: spatial and channel-shared**

b) **DY-ReLU-B: spatial-shared and channel-wise**

c) **DY-ReLU-C: spatial and channel-wise**

计算量/表示能力 increase

Implementation of hyper function $\theta(x)$

$$a_c^k(x) = \alpha^k + \lambda_a \Delta a_c^k(x)$$

$$b_c^k(x) = \beta^k + \lambda_b \Delta b_c^k(x)$$

α^k and β^k are initialization values of a_c^k and b_c^k

λ_a and λ_b are scalars that control the range of residual

Mobile sub-block

Experimental Results of dynamic relu:

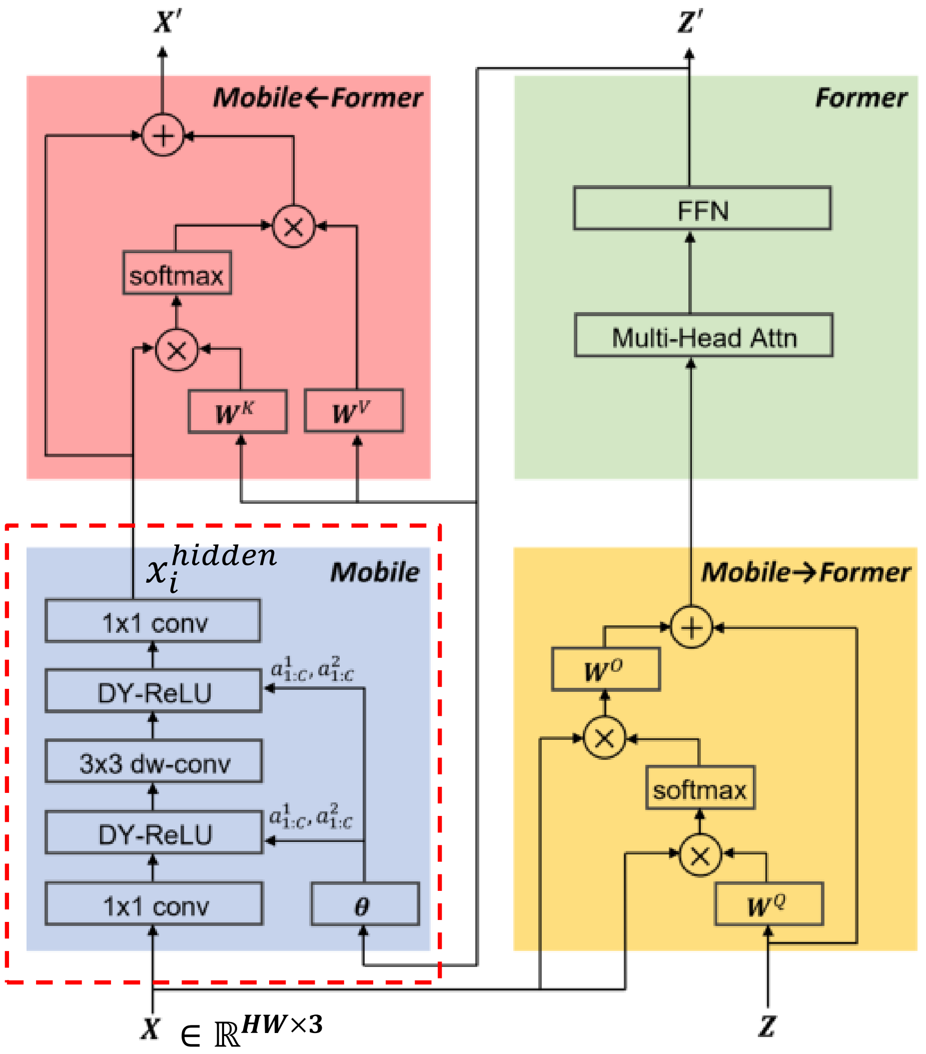
Network	Activation	#Param	MAdds	Top-1	Top-5
MobileNetV2 ×1.0	ReLU	3.5M	300.0M	72.0	91.0
	DY-ReLU	7.5M	315.5M	$76.2_{(4.2)}$	$93.1_{(2.1)}$
MobileNetV2 ×0.75	ReLU	2.6M	209.0M	69.8	89.6
	DY-ReLU	5.0M	221.7M	$74.3_{(4.5)}$	$91.7_{(2.1)}$
MobileNetV2 ×0.5	ReLU	2.0M	97.0M	65.4	86.4
	DY-ReLU	3.1M	104.5M	$70.3_{(4.9)}$	$89.3_{(2.9)}$
MobileNetV2 ×0.35	ReLU	1.7M	59.2M	60.3	82.9
	DY-ReLU	2.7M	65.0M	$66.4_{(6.1)}$	$86.5_{(3.6)}$
MobileNetV3-Large	ReLU/SE/HS	5.4M	219.0M	75.2	92.2
	DY-ReLU	9.8M	230.5M	$75.9_{(0.7)}$	$92.7_{(0.5)}$
MobileNetV3-Small	ReLU/SE/HS	2.9M	66.0M	67.4	86.4
	DY-ReLU	4.0M	68.7M	$69.7_{(2.3)}$	$88.3_{(1.9)}$
ResNet-50	ReLU	23.5M	3.86G	76.2	92.9
	DY-ReLU	27.6M	3.88G	$77.2_{(1.0)}$	$93.4_{(0.5)}$
ResNet-34	ReLU	21.3M	3.64G	73.3	91.4
	DY-ReLU	24.5M	3.65G	$74.4_{(1.1)}$	$92.0_{(0.6)}$
ResNet-18	ReLU	11.1M	1.81G	69.8	89.1
	DY-ReLU	12.8M	1.82G	$71.8_{(2.0)}$	$90.6_{(1.5)}$
ResNet-10	ReLU	5.2M	0.89G	63.0	84.7
	DY-ReLU	6.3M	0.90G	$66.3_{(3.3)}$	$86.7_{(2.0)}$

Table 4. Comparing DY-ReLU with baseline activation functions (ReLU, SE or h-swish, denoted as HS) on ImageNet [5] classification in three network architectures. DY-ReLU-B with $K = 2$ linear functions is used. Note that SE blocks are removed when using DY-ReLU in MobileNetV3. The numbers in brackets denote the performance improvement over the baseline. DY-ReLU outperforms its counterpart for all networks.

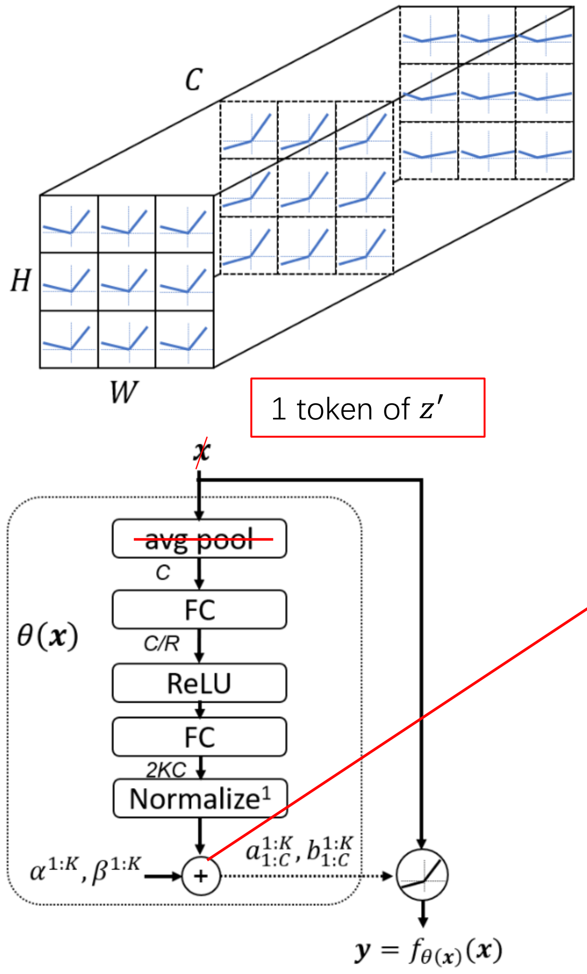
Activation	K	MobileNetV2 ×0.35 #Param	MAdds	Top-1	MobileNetV2 ×1.0 #Param	MAdds	Top-1
ReLU	2	1.7M	59.2M	60.3	3.5M	300.0M	72.0
RReLU [40]	2	1.7M	59.2M	$60.0_{(-0.3)}$	3.5M	300.0M	$72.5_{(+0.5)}$
LeakyReLU [25]	2	1.7M	59.2M	$60.9_{(+0.6)}$	3.5M	300.0M	$72.7_{(+0.7)}$
PReLU [10]	2	1.7M	59.2M	$63.1_{(+2.8)}$	3.5M	300.0M	$73.3_{(+1.3)}$
SE[14]+ReLU	2	2.1M	62.0M	$62.8_{(+2.5)}$	5.1M	307.5M	$74.2_{(+2.2)}$
Maxout [7]	2	2.1M	106.6M	$64.9_{(+4.6)}$	5.7M	575.8M	$75.1_{(+3.1)}$
Maxout [7]	3	2.4M	157.6M	$65.4_{(+5.1)}$	7.8M	860.2M	$75.8_{(+3.8)}$
DY-ReLU-B	2	2.7M	65.0M	$66.4_{(+6.1)}$	7.5M	315.5M	$\mathbf{76.2}_{(+4.2)}$
DY-ReLU-B	3	3.1M	67.8M	$\mathbf{66.6}_{(+6.3)}$	9.2M	322.8M	$\mathbf{76.2}_{(+4.2)}$

Table 5. Comparing DY-ReLU with related activation functions on ImageNet [5] classification. MobileNetV2 with width multiplier ×0.35 and ×1.0 are used. We use spatial-shared and channel-wise DY-ReLU-B with $K = 2, 3$ linear functions. The numbers in brackets denote the performance improvement over the baseline. DY-ReLU outperforms all prior work including Maxout, which has significantly more computations.

Mobile sub-block



($Z \in$ 6×192 in 294M FLOPs)



DY-ReLU-B: spatial-shared and channel-wise

$$a_c^k(x) = \alpha^k + \lambda_a \Delta a_c^k(x)$$

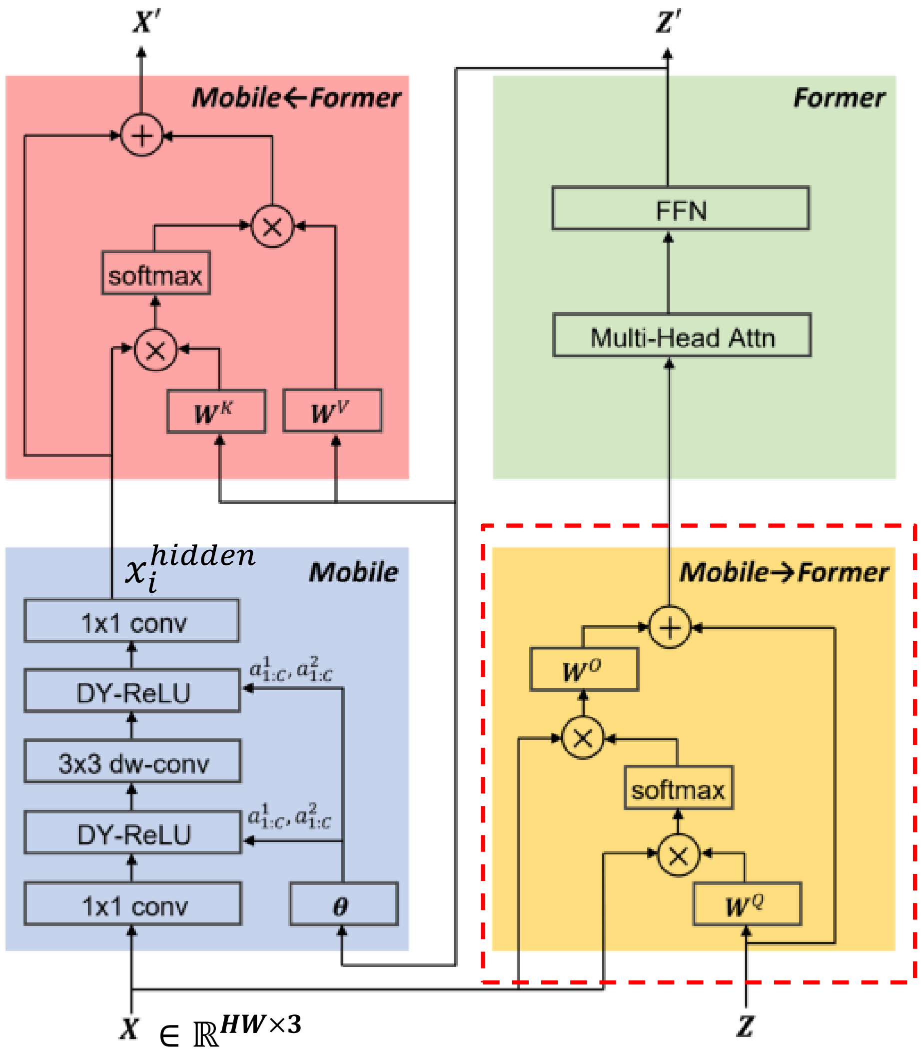
$$b_c^k(x) = \beta^k + \lambda_b \Delta b_c^k(x)$$

α^k and β^k are initialization values of a_c^k and b_c^k

λ_a and λ_b are scalars that control the range of residual

Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z. (2020). Dynamic ReLU. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science(), vol 12364. Springer, Cham.

Mobile→Former sub-block



($Z \in$ 6×192 in 294M FLOPs)

The projection matrices for the key and value are removed from Mobile side

Low Cost Two-way Bridge:

Local feature map $X \in \mathbb{R}^{HW\times 3}$, global tokens $Z \in \mathbb{R}^{M\times d}$ (M ≤ 6)

The light-weight cross attention from local feature map X to global tokens Z is computed as:

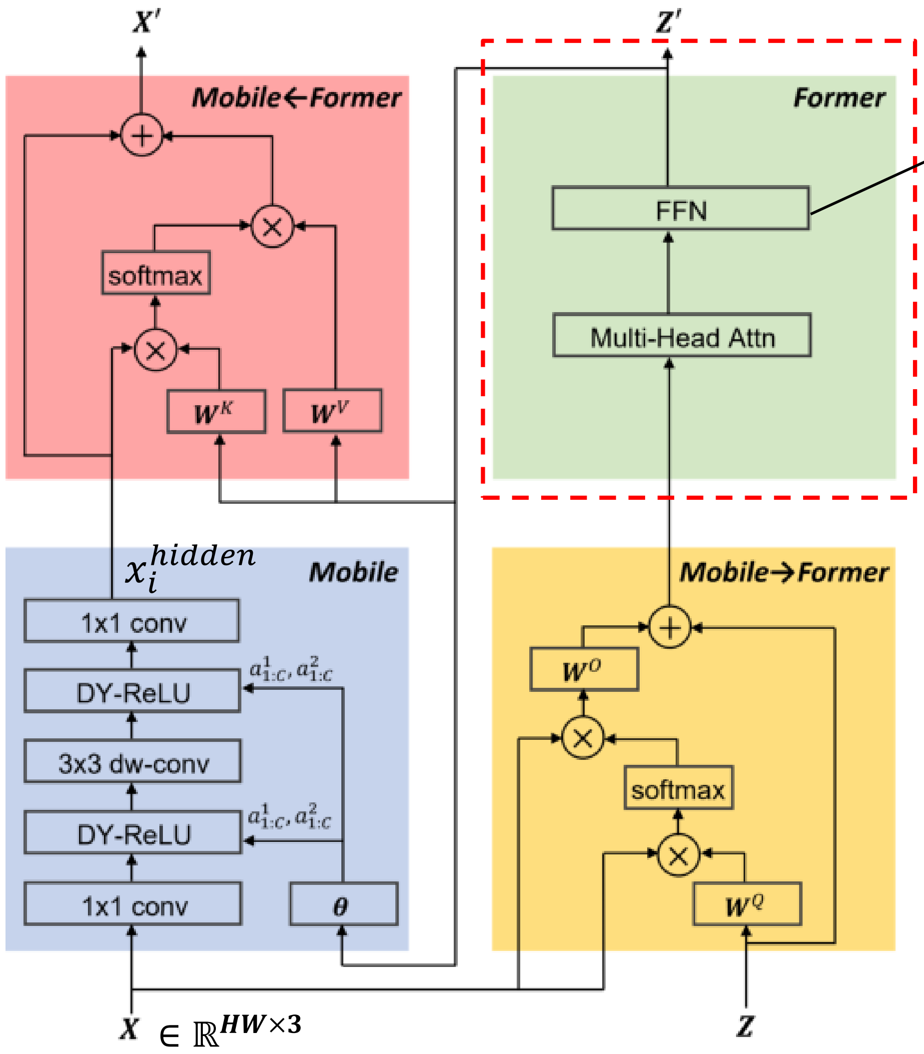
$$A_X \rightarrow Z = [Attn(\tilde{z}_i W_i^Q, \tilde{x}_i, \tilde{x}_i)]_{i=1:h} W^O \quad (\tilde{z}_i \in \mathbb{R}^{M\times \frac{d}{h}})$$

$$\text{h heads as } X = [\tilde{x}_1, \cdots \tilde{x}_h], \quad Z = [\tilde{z}_1, \cdots \tilde{z}_h]$$

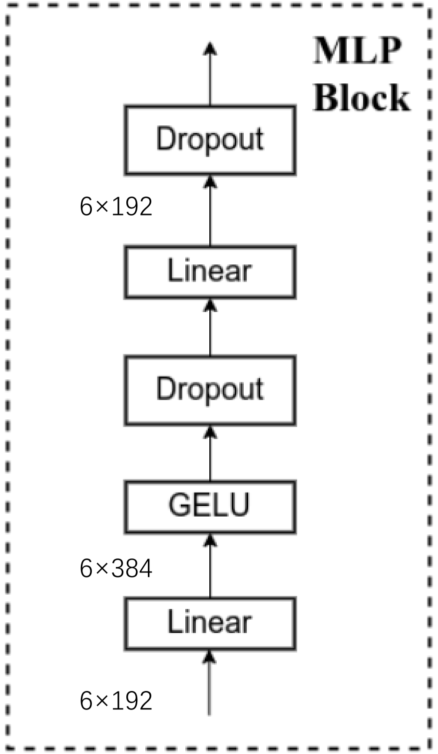
$$Attn(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

W_i^Q projection matrices for the i^{th} head.
W^O is used to combine multiple heads together.
$[\cdot]_{1:h}$ denotes the concatenation of h

Former sub-block

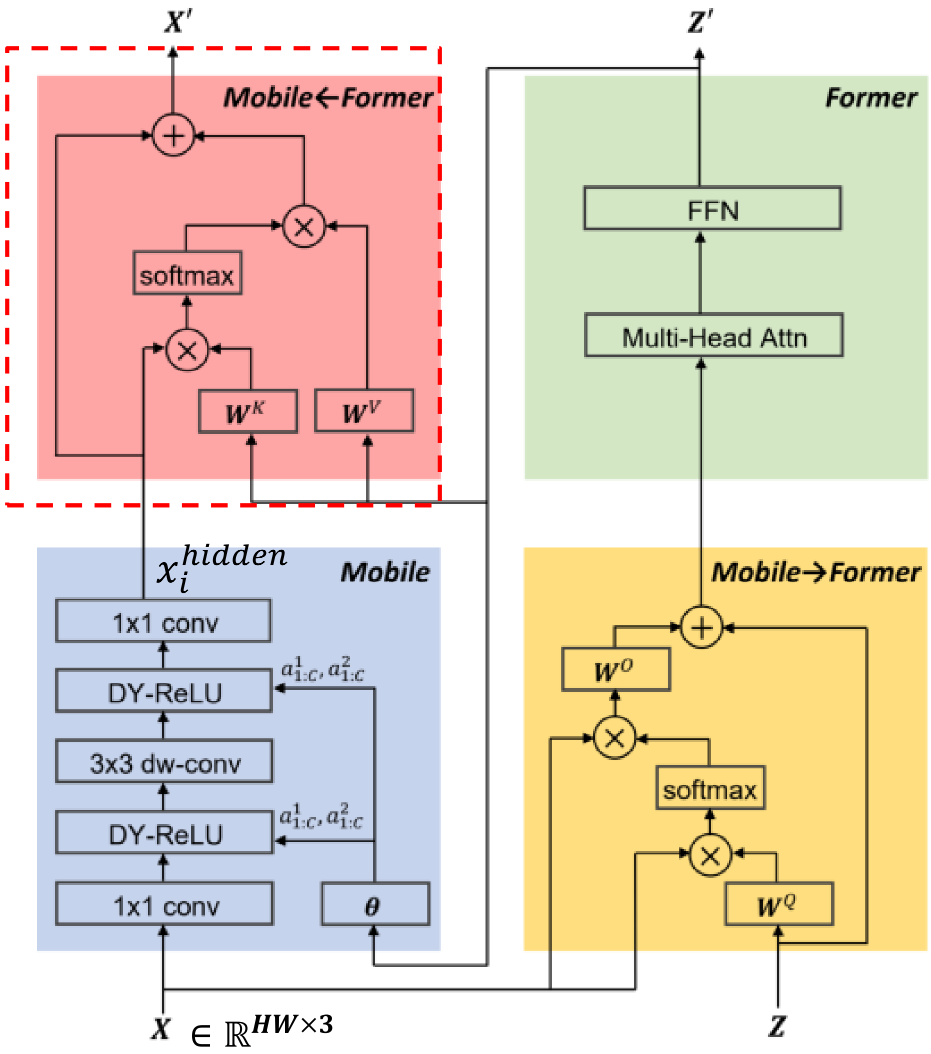


Expansion ratio 2 (instead of 4) is used if FFN



($Z \in$ 6×192 in 294M FLOPs)

Former → Mobile sub-block



($Z \in$ 6×192 in 294M FLOPs)

The projection matrix of the query

is removed from Mobile side.

Low Cost Two-way Bridge:

Local feature map $X \in \mathbb{R}^{HW\times 3}$, global tokens $Z \in \mathbb{R}^{M\times d}$ (M ≤ 6)

the cross attention from global to local is computed as:

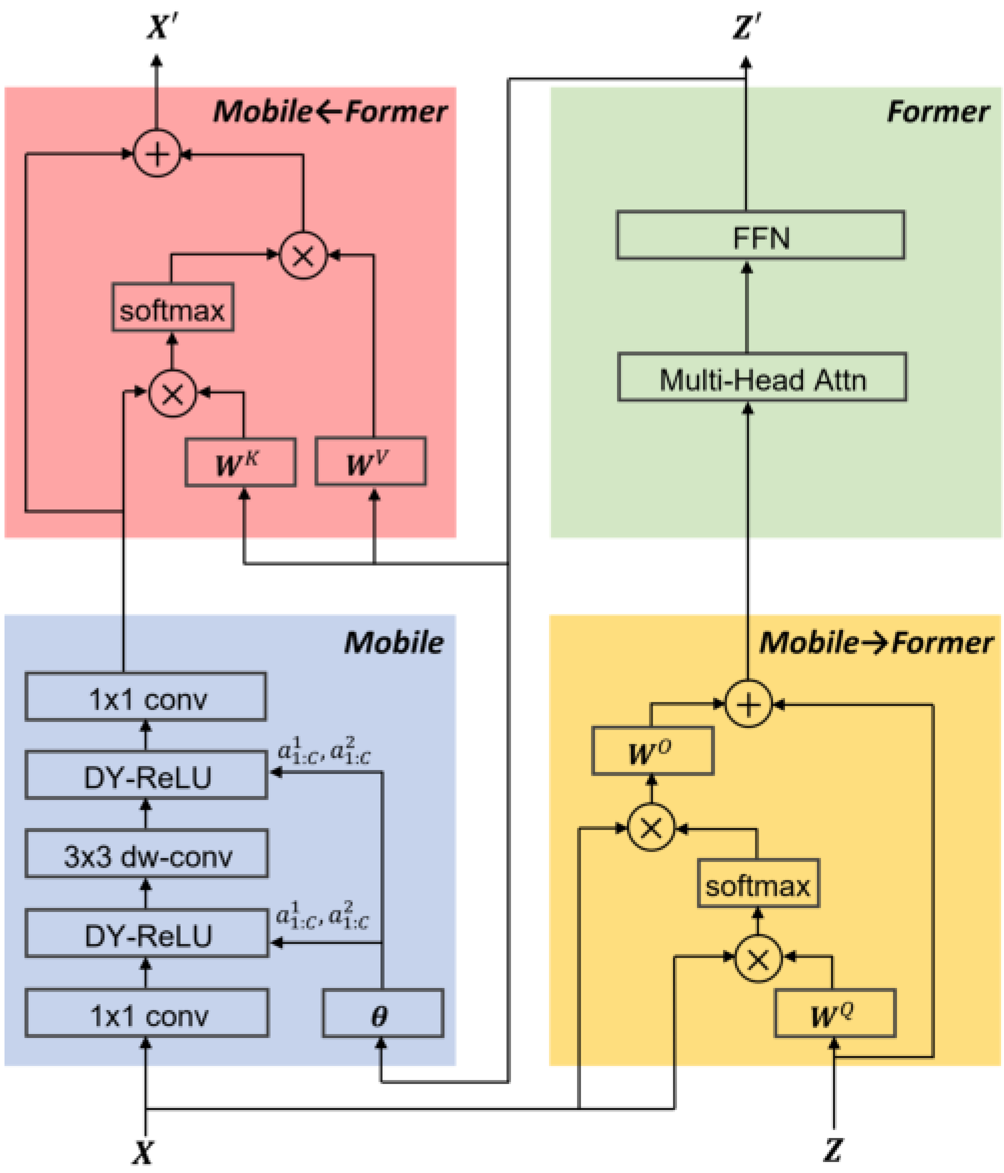
$$A_Z \rightarrow \mathrm{X} = [Attn(\tilde{x}_i, \tilde{z}_i W_i^K, \tilde{z}_i W_i^V)]_{i=1:h} \quad (\tilde{z}_i \in \mathbb{R}^{M\times \frac{d}{h}})$$

$$\text{h heads as } X = [\tilde{x}_1, \cdots \tilde{x}_h], \quad \mathrm{Z} = [\tilde{z}_1, \cdots \tilde{z}_h]$$

$$Attn(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$[\cdot]_{1:h}$ denotes the concatenation of h

Overview



$X \in \mathbb{R}^{HW \times 3}$

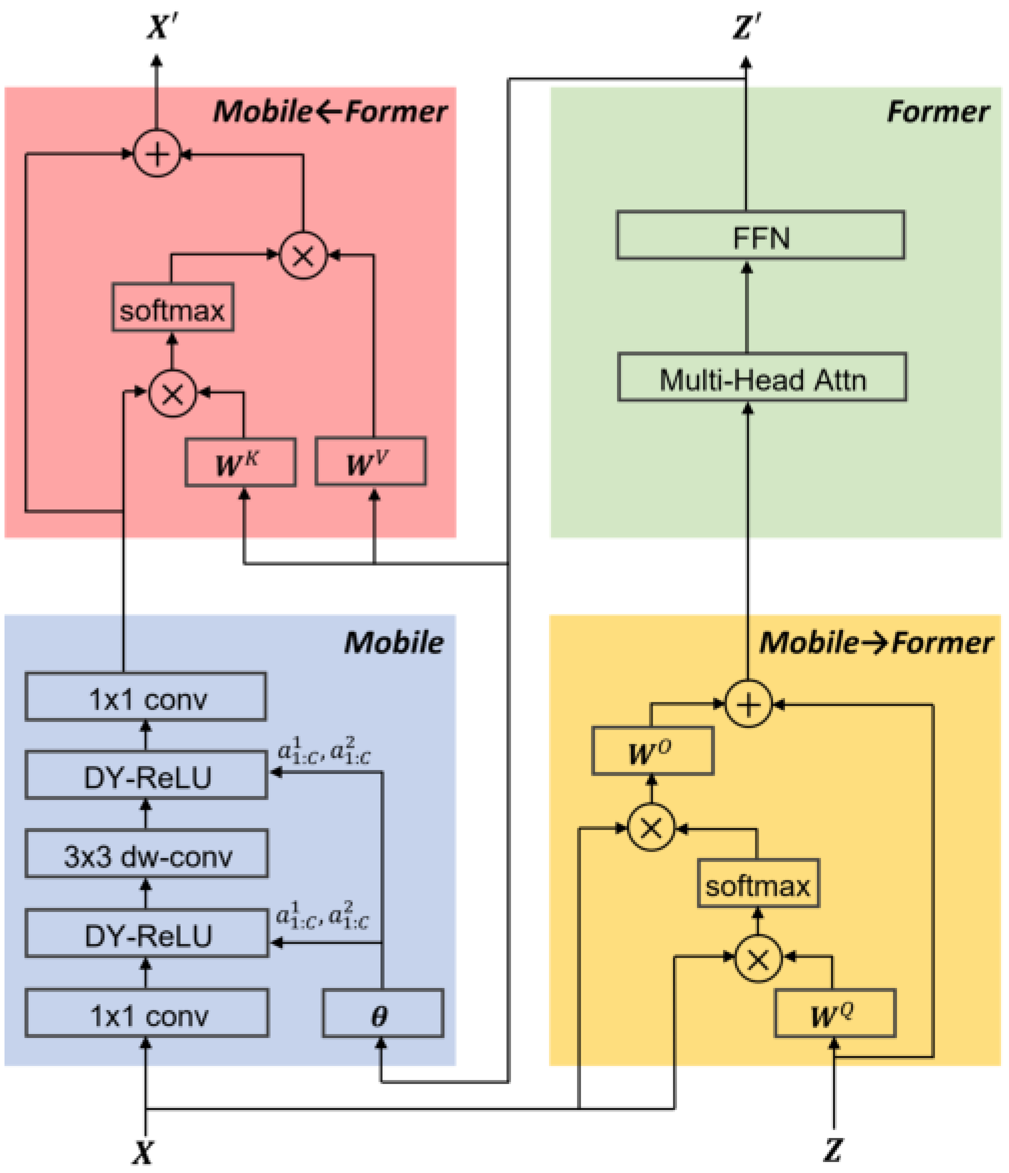
$Z \in \mathbb{R}^{M \times d}$ $(M \leq 6)$

- **Light-weight**

1. Performe the cross attention at the bottleneck of Mobile where the number of channels is low.
2. Remove projections on query,key and value(W^Q,W^K,W^V)from Mobile side.
3. Significantly fewer tokens ($M \leq 6$) randomly initialized
4. Save the average pooling by applying the two MLP layers on the first global token output $Z_1'$from Former

The bridge and Former consume less than 20% of the total computational cost,but significantly improve the representation capability.

Mobile-Former architecture with 294M FLOPs for image size 224×224



$X \in \mathbb{R}^{HW\times 3}$ $\quad$ $Z \in \mathbb{R}^{M\times d}$ (M ≤ 6)

Stage	Input	Operator	exp size	#out	Stride
tokens	6×192	–	–	–	–
stem	$224^2\times3$	conv2d, 3×3	–	16	2
1	$112^2\times16$	MicroNet bneck-lite	32	16	1
2	$112^2\times16$	Mobile-Former↓	96	24	2
	$56^2\times24$	Mobile-Former	96	24	1
3	$56^2\times24$	Mobile-Former↓	144	48	2
	$28^2\times48$	Mobile-Former	192	48	1
4	$28^2\times48$	Mobile-Former↓	288	96	2
	$14^2\times96$	Mobile-Former	384	96	1
	$14^2\times96$	Mobile-Former	576	128	1
	$14^2\times128$	Mobile-Former	768	128	1
5	$14^2\times128$	Mobile-Former↓	768	192	2
	$7^2\times192$	Mobile-Former	1152	192	1
	$7^2\times192$	Mobile-Former	1152	192	1
	$7^2\times192$	conv2d, 1×1	–	1152	1
head	$7^2\times1152$	pool, 7×7	–	1152	–
	$1^2\times1152$	concat w/ cls token	–	1344	–
	$1^2\times1344$	FC	–	1920	–
	$1^2\times1920$	FC	–	1000	–

Table 1. **Specification for Mobile-Former-294M.** "bneck-lite" denotes the lite bottleneck block. "Mobile-Former↓" denotes the variant of downsample block.

Adapting position embedding in head

The feature and position q_k^f and q_k^p

$$q_{k+1}^p = q_k^p + g(q_{k+1}^f)$$

Spatial-aware dynamic ReLU in backbone

$\theta = f(z_1)$,where z_1 is the first global token

θ_i per spatial position i in a feature map

$$\theta_i = \sum_j \alpha_{i,j} f(z_j), s.t. \sum_j \alpha_{i,j} = 1$$

Outperforms DETR by 1.1 AP but saves 52% of computational cost (41G vs. 86G) and 36% of parameters. (26.6M vs. 41.3M)

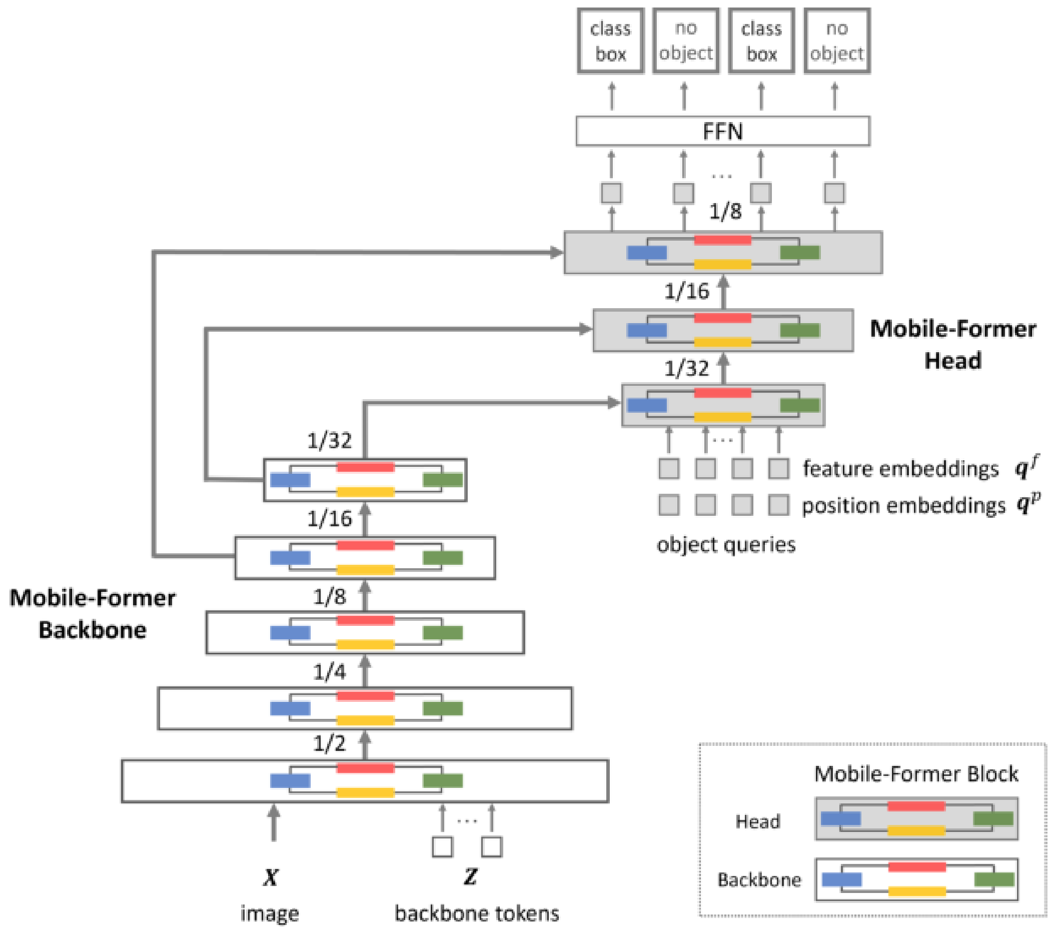


Figure 4. **Mobile-Former for object detection**. Both backbone and head use Mobile-Former blocks (see Figure 1, 3). The backbone has 6 global tokens while the head has 100 object queries. All object queries pass through multiple resolutions ($\frac{1}{32}$, $\frac{1}{16}$, $\frac{1}{8}$) in the head. Similar to DETR [1], feed forward network (FFN) is used to predict class label and bounding box. Best viewed in color.

Experiments – Image Classification on ImageNet

Model	FLOPs	TOP-1
MobileNetV3	356M	76.6
LeViT	305M	76.6
Mobile-Former (ours)	**294M**	**77.9**

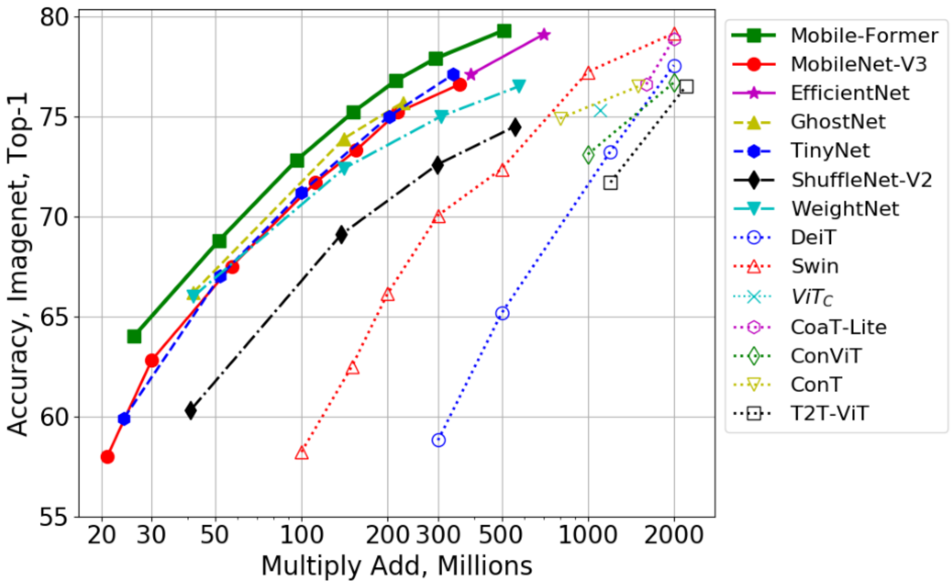


Figure 2. **Comparison among Mobile-Former, efficient CNNs and vision transformers**, in terms of accuracy over FLOPs. The comparison is performed on ImageNet classification. Mobile-Former consistently outperforms both efficient CNNs and vision transformers in low FLOP regime (from 25M to 500M MAdds). Note that we implement Swin [22] and DeiT [32] at low computational budget from 100M to 2G FLOPs. Best viewed in color.

Mobile-Former achieves 77.9% top-1 accuracy at 294M FLOPs, outperforming MobileNetV3 and LeViT by a clear margin

When the size of the entered picture is large, Mobile-Former consistently outperforms both efficient CNNs and vision transformers from 25M to 500M FLOPs. Showcasing the usage of transformer at the low FLOP regime where efficient CNNs dominate.

Experiments – Image Classification on ImageNet (Ablation)

Model	#Params	MAdds	Top-1	Top-5
Mobile (using ReLU)	6.1M	259M	74.2	91.8
+ *Former* and Bridge	10.1M	290M	$76.8_{(+2.6)}$	$93.2_{(+1.4)}$
+ DY-ReLU in *Mobile*	11.4M	294M	$77.8_{(+1.0)}$	$93.7_{(+0.5)}$

Table 4. **Ablation of Former+bridge and dynamic ReLU** evaluated on ImageNet classification.Mobile-Former-294M is used.

#Tokens	#Params	MAdds	Top-1	Top-5
1	11.4M	269M	77.1	93.2
3	11.4M	279M	77.6	93.6
6	11.4M	294M	77.8	93.7
9	11.4M	309M	77.7	93.8

Table 5. **Ablation of the number of tokens** on ImageNet classification.Mobile-Former-294M is used.

Token Dimension	#Params	MAdds	Top-1	Top-5
64	7.3M	277M	76.8	93.1
128	9.1M	284M	77.3	93.5
192	11.4M	294M	77.8	93.7
256	14.3M	308M	77.8	93.7
320	17.9M	325M	77.6	93.6

Table 6. **Ablation of token dimension** on ImageNet classification. Mobile-Former-294M is used.

Attention	FFN	#Params	MAdds	Top-1	Top-5
MHA	✓	11.4M	294M	77.8	93.7
MHA	✗	9.8M	284M	77.5	93.6
Pos-Mix-MLP	✓	10.5M	284M	77.3	93.5

Table 7. **Ablation of multi-head attention(MHA) and FFN** on ImageNet classification.Mobile-Former-294M is used.

Experiments – Image Classification on ImageNet

Model	Input	#Params	MAdds	Top-1
MobileNetV3 Small 1.0× [15]	160^2	2.5M	30M	62.8
Mobile-Former-26M	224^2	3.2M	**26M**	**64.0**
MobileNetV3 Small 1.0× [15]	224^2	2.5M	57M	67.5
Mobile-Former-52M	224^2	3.5M	**52M**	**68.7**
MobileNetV3 1.0× [15]	160^2	5.4M	112M	71.7
Mobile-Former-96M	224^2	4.6M	**96M**	**72.8**
ShuffleNetV2 1.0× [25]	224^2	2.2M	**138M**	69.1
ShuffleNetV2 1.0×+WeightNet 4× [24]	224^2	5.1M	141M	72.4
MobileNetV3 0.75× [15]	224^2	4.0M	155M	73.3
Mobile-Former-151M	224^2	7.6M	151M	**75.2**
MobileNetV3 1.0× [15]	224^2	5.4M	217M	75.2
Mobile-Former-214M	224^2	9.4M	**214M**	**76.7**
ShuffleNetV2 1.5× [25]	224^2	3.5M	299M	72.6
ShuffleNetV2 1.5×+WeightNet 4× [24]	224^2	9.6M	307M	75.0
MobileNetV3 1.25× [15]	224^2	7.5M	356M	76.6
EfficientNet-B0 [28]	224^2	5.3M	390M	77.1
Mobile-Former-294M	224^2	11.4M	**294M**	**77.9**
ShuffleNetV2 2× [25]	224^2	5.5M	557M	74.5
ShuffleNetV2 2×+WeightNet 4× [24]	224^2	18.1M	573M	76.5
Mobile-Former-508M	224^2	14.0M	**508M**	**79.3**

Tabel 2. Comparing Mobile-Former with efficient CNNs evaluated on ImageNet classification.

Model	Input	#Params	MAdds	Top-1
T2T-ViT-7 [44]	224^2	4.3M	1.2G	71.7
DeiT-Tiny [32]	224^2	5.7M	1.2G	72.2
ConViT-Tiny [6]	224^2	6.0M	1.0G	73.1
ConT-Ti [42]	224^2	5.8M	0.8G	74.9
ViT_C [40]	224^2	4.6M	1.1G	75.3
ConT-S [42]	224^2	10.1M	1.5G	76.5
Swin-1G [22] ‡	224^2	7.3M	1.0G	77.3
Mobile-Former-294M	224^2	11.4M	**294M**	**77.9**
PVT-Tiny [37]	224^2	13.2M	1.9G	75.1
T2T-ViT-12 [44]	224^2	6.9M	2.2G	76.5
CoaT-Lite Tiny [41]	224^2	5.7M	1.6G	76.6
ConViT-Tiny+ [6]	224^2	10.0M	2G	76.7
DeiT-2G [32] ‡	224^2	9.5M	2.0G	77.6
CoaT-Lite Mini [41]	224^2	11.0M	2.0G	78.9
BoT-S1-50 [27]	224^2	20.8M	4.3G	79.1
Swin-2G [22] ‡	224^2	12.8M	2.0G	79.2
Mobile-Former-508M	224^2	14.0M	**508M**	**79.3**

Tabel 3. Comparing Mobile-Former with vision transformer variants evaluated on ImageNet classification.

Experiments – Object Detection On COCO

Model	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	MAdds (G)	#Params (M)
Shuffle-V2 [25]	25.9	41.9	26.9	12.4	28.0	36.4	**2.6** (161)	0.8 (10.4)
MF-151M	**34.2**	53.4	36.0	19.9	36.8	45.3	**2.6** (161)	4.9 (14.4)
Mobile-V3 [15]	27.2	43.9	28.3	13.5	30.2	37.2	4.7 (162)	2.8 (12.3)
MF-214M	**35.8**	55.4	38.0	21.8	38.5	46.8	**3.9** (162)	5.7 (15.2)
ResNet18 [13]	31.8	49.6	33.6	16.3	34.3	43.2	29 (181)	11.2 (21.3)
MF-294M	**36.6**	56.6	38.6	21.9	39.5	47.9	**5.5** (164)	6.5 (16.1)
ResNet50 [13]	36.5	55.4	39.1	20.4	40.3	48.1	84 (239)	23.3 (37.7)
PVT-Tiny [37]	36.7	56.9	38.9	22.6	38.8	50.0	70 (221)	12.3 (23.0)
ConT-M [42]	37.9	58.1	40.2	23.0	40.6	50.4	65 (217)	16.8 (27.0)
MF-508M	**38.0**	58.3	40.3	22.9	41.2	49.7	**9.8** (168)	8.4 (17.9)

Table 8. **COCO object detection results in RetinaNet framework**. All models are trained on train2017 for 12 epochs(1×)from ImageNet pretrained weights, and tested on val2017. We use initial MF(e.g. MF-508M) to refer Mobile-Former. Madds and #Params are in the format of "backbone(total)". MAdds is based on the image size 800×1333.

Model	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	MAdds (G)	#Params (M)
DETR [1]	42.0	62.4	44.2	20.5	45.8	61.1	86	41.3
DETR-DC5 [1]	**43.3**	**63.1**	45.9	22.5	**47.3**	**61.1**	187	41.3
E2E-MF-508M	43.1	61.9	**46.8**	**23.8**	46.5	60.4	**41.4**	**26.6**
E2E-MF-294M	40.5	58.8	43.5	20.6	44.0	56.9	**24.1**	**25.1**
E2E-MF-214M	39.3	57.3	42.1	19.9	42.4	56.6	**17.8**	**20.1**
E2E-MF-151M	37.2	54.5	39.9	17.4	39.8	54.9	**12.7**	**14.8**

Table 9. **End-to-end object detection results on COCO**.All models are trained on train2017 and tested on val2017.DETR baselines are trained for 500 epochs,while our Mobile-Former models are trained for 300 epochs. We use initial E2EMF(e.g. E2E-MF-508M) to refer end-to-end Mobile-Former detectors.Madds is based on image size 800×1333.

Finally we note that *exploring the optimal network parameters (e.g. width, height) in Mobile-Former is not a goal of this work*, rather we demonstrate that the parallel design provides an efficient and effective network architecture.

Y. Chen et al., "Mobile-Former: Bridging MobileNet and Transformer," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 5260-5269, doi: 10.1109/CVPR52688.2022.00520.

感 谢 聆 听

汇报人：张思源