An Image Patch is a Wave: Phase-Aware Vision MLP (Supplementary Material)

Yehui Tang[1,2], Kai Han[2], Jianyuan Guo[2,3], Chang Xu[3],
Yanxi Li[2,3], Chao Xu[1], Yunhe Wang[2*]
[1]School of Artificial Intelligence, Peking University. [2]Huawei Noah's Ark Lab.
[3]School of Computer Science, University of Sydney.
yhtang@pku.edu.cn, {kai.han, yunhe.wang}@huawei.com.

1. Detailed Architectures

Table 1 shows the detailed specifications of the proposed Wave-MLP architecture. To get hierarchical features, we split the whole model into four stages, and reduce the size of feature map stage-wisely. The Wave-MLP family contains four models with different parameters and computational costs by adjusting the depths and widths of architecture specifications, which are denoted as Wave-MLP-T, Wave-MLP-S, Wave-MLP-M, and Wave-MLP-B, sequentially. From Wave-MLP-T to Wave-MLP-B, the number of parameters varies from 17M to 63M, and FLOPs varies from 2.4G to 10.2G.

2. More Experiments

For the object detection and instance segmentation tasks on COCO [3], we further train Mask R-CNN models with 3× schedule and multi-scale training strategy [1]. The results of different backbone are shown in Table 2. Compared with other backbones, the proposed Wave-MLP achieves much higher performance. For example, our Wave-MLP-T achieves 44.1 box AP and 40.1 mask AP with 25.3M parameters and 196.3G FLOPs, which is significantly superior to the PVT-Tiny model with 39.8 box AP, 37.4 mask AP, 32.9M parameters and 208.1G FLOPs.

References

[1] 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. 1, 2

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[3] 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. 1

[4] 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 *IEEE ICCV*, 2021. 2

[*] Corresponding author.

Table 1. Detailed architecture specifications of Wave-MLP. 'Dimension' and 'expansion' denote the dimension of feature and expand ratio, respectively. H and W are the height and width of input image. FLOPs is calculated with input size of 224×224.

	Output size	Wave-MLP-T	Wave-MLP-S	Wave-MLP-M	Wave-MLP-B
stage 1	$\frac{H}{4} \times \frac{W}{4}$	$\begin{bmatrix}\text{dimension} = 64\\ \text{expansion} = 4\end{bmatrix} \times 2$	$\begin{bmatrix}\text{dimension} = 64\\ \text{expansion} = 4\end{bmatrix} \times 2$	$\begin{bmatrix}\text{dimension} = 64\\ \text{expansion} = 8\end{bmatrix} \times 3$	$\begin{bmatrix}\text{dimension} = 96\\ \text{expansion} = 4\end{bmatrix} \times 2$
stage 2	$\frac{H}{8} \times \frac{W}{8}$	$\begin{bmatrix}\text{dimension} = 128\\ \text{expansion} = 4\end{bmatrix} \times 2$	$\begin{bmatrix}\text{dimension} = 128\\ \text{expansion} = 4\end{bmatrix} \times 3$	$\begin{bmatrix}\text{dimension} = 128\\ \text{expansion} = 8\end{bmatrix} \times 4$	$\begin{bmatrix}\text{dimension} = 192\\ \text{expansion} = 4\end{bmatrix} \times 2$
stage 3	$\frac{H}{16} \times \frac{W}{16}$	$\begin{bmatrix}\text{dimension} = 320\\ \text{expansion} = 4\end{bmatrix} \times 4$	$\begin{bmatrix}\text{dimension} = 320\\ \text{expansion} = 4\end{bmatrix} \times 10$	$\begin{bmatrix}\text{dimension} = 320\\ \text{expansion} = 4\end{bmatrix} \times 18$	$\begin{bmatrix}\text{dimension} = 384\\ \text{expansion} = 4\end{bmatrix} \times 18$
stage 4	$\frac{H}{32} \times \frac{W}{32}$	$\begin{bmatrix}\text{dimension} = 512\\ \text{expansion} = 4\end{bmatrix} \times 2$	$\begin{bmatrix}\text{dimension} = 512\\ \text{expansion} = 4\end{bmatrix} \times 3$	$\begin{bmatrix}\text{dimension} = 512\\ \text{expansion} = 4\end{bmatrix} \times 3$	$\begin{bmatrix}\text{dimension} = 768\\ \text{expansion} = 4\end{bmatrix} \times 2$
# Parameters		17M	30M	44M	63M
FLOPs		2.4G	4.5G	7.9G	10.2G

Table 2. Results of object detection and instance segmentation on COCO val2017. The Mask R-CNN model trained with 3× schedule and multi-scale training strategy [1] is used as the detector.

Backbone	Params. / FLOPs	AP^{b}	$\text{AP}^{\text{b}}_{50}$	$\text{AP}^{\text{b}}_{75}$	AP^{b}_{S}	AP^{b}_{M}	AP^{b}_{L}	AP^{m}	$\text{AP}^{\text{m}}_{50}$	$\text{AP}^{\text{m}}_{75}$	AP^{m}_{S}	AP^{m}_{M}	AP^{m}_{L}
ResNet18 [2]	31.2M / 207.3G	36.9	57.1	40.0	-	-	-	33.6	53.9	35.7	-	-	-
PVT-Tiny [4]	32.9M / 208.1G	39.8	62.2	43.0	-	-	-	37.4	59.3	39.9	-	-	-
Wave-MLP-T	25.3M / 196.3G	**44.1**	66.0	48.2	28.4	47.6	55.9	**40.1**	63.1	43.2	24.3	43.5	53.2
ResNet50 [2]	44.2M / 260.1G	41.0	61.7	44.9	-	-	-	37.1	58.4	40.1	-	-	-
PVT-Small [4]	44.1M / 245.1G	43.0	65.3	46.9	-	-	-	39.9	62.5	42.8	-	-	-
Wave-MLP-S	37.1M / 231.3G	**45.5**	66.9	49.3	29.4	48.7	58.7	**41.0**	64.2	44.0	25.0	44.2	54.7
ResNet101 [2]	63.2M / 336.4G	42.8	63.2	-	-	-	47.1	38.5	60.1	41.3	-	-	-
PVT-Medium	63.9M / 301.7G	44.2	66.0	48.2	-	-	-	40.5	63.1	43.5	-	-	-
PVT-Large	71.1M / 345.7G	44.5	66.0	48.3	-	-	-	40.7	63.4	43.7	-	-	-
Wave-MLP-M	49.4M / 291.3G	**46.3**	67.8	50.3	29.5	49.3	60.3	**41.5**	65.2	44.1	24.9	44.7	55.6