RaMLP: Vision MLP via Region-aware Mixing

Shenqi Lai[1], **Xi Du**[2], **Jia Guo**[1] and **Kaipeng Zhang**[3]
[1]InsightFace.ai
[2]Kiwi Tech
[3]Shanghai AI Laboratory
laishenqi@qq.com, leo.du@kiwiar.com, guojia@gmail.com, kp_zhang@foxmail.com

Abstract

Recently, MLP-based architectures achieved impressive results in image classification against CNNs and ViTs. However, there is an obvious limitation in that their parameters are related to image sizes, allowing them to process only fixed image sizes. Therefore, they cannot directly adapt dense prediction tasks (e.g., object detection and semantic segmentation) where images are of various sizes. Recent methods tried to address it but brought two new problems, long-range dependencies or important visual cues are ignored. This paper presents a new MLP-based architecture, Region-aware MLP (RaMLP), to satisfy various vision tasks and address the above three problems. In particular, we propose a well-designed module, Region-aware Mixing (RaM). RaM captures important local information and further aggregates these important visual clues. Based on RaM, RaMLP achieves a global receptive field even in one block. It is worth noting that, unlike most existing MLP-based architectures that adopt the same spatial weights to all samples, RaM is region-aware and adaptively determines weights to extract region-level features better. Impressively, our RaMLP outperforms state-of-the-art ViTs, CNNs, and MLPs on both ImageNet-1K image classification and downstream dense prediction tasks, including MS-COCO object detection, MS-COCO instance segmentation, and ADE20K semantic segmentation. In particular, RaMLP outperforms MLPs by a large margin (around 1.5% Apb or 1.0% mIoU) on dense prediction tasks. The training code could be found at https://github.com/xiaolai-sqlai/RaMLP.

1 Introduction

In the past decade, Convolutional Neural Networks (CNNs) [Krizhevsky *et al.*, 2012] have shown great success in various computer vision tasks. In recent years, transformers trained by large-scale data [Devlin *et al.*, 2019] dominate most natural language processing tasks. Motivated by that, many research works proposed Vision Transformers (ViTs) [Dosovitskiy *et al.*, 2021; Touvron *et al.*, 2021b; Liu *et al.*, 2021; Yang *et al.*, 2021; Wu *et al.*, 2021a; Wu *et al.*, 2021b; Yuan *et al.*, 2021; Wang *et al.*, 2021], the transformer-based architectures specific for vision, and surpassed CNNs when using large-scale training data.

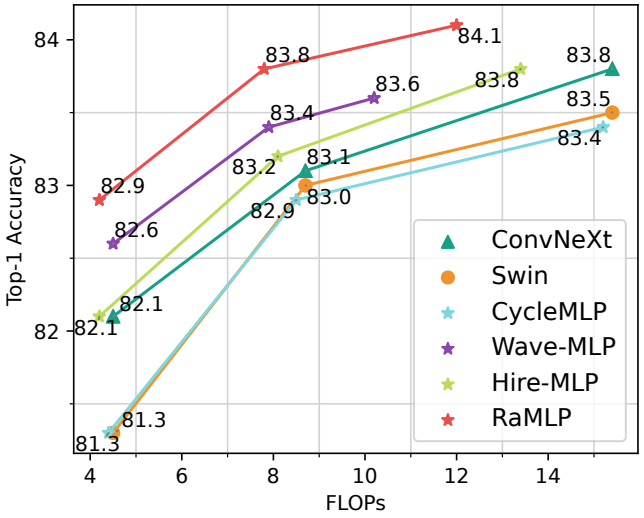

Figure 1: Results of different models on ImageNet-1K validation set. Comparing the performance and FLOPs of recent models ConvNeXt, Swin Transformer, CycleMLP, Hire-MLP, Wave-MLP, and our RaMLP. Triangle means the CNNs, circle means the ViTs, and star means the MLPs.

More recently, MLP-Mixer [Tolstikhin *et al.*, 2021] was proposed to prove the potential of MLPs. Its parameters are almost all learned from fully-connected layers. It achieved comparable results in image classification against CNN-based or ViT-based architectures. Such promising results drive some exploration of MLP-based architectures.

Followed by MLP-Mixer, many advanced MLP-based architectures [Touvron *et al.*, 2021a; Guo *et al.*, 2021; Hou *et al.*, 2021; Wang *et al.*, 2022b] were proposed last year, and they achieved more impressive results in image classification even surpassing CNN-based or ViT-based architectures. However, they cannot be transferred to dense prediction tasks (e.g., object detection and semantic segmentation) since their parameters are image sizes related and can not cope with images from various image sizes. Specifically, the global recep-

tive field is crucial in computer vision tasks, and they obtain it through matrix transposition and token-mixing projection such that the long-range dependencies are covered. However, this operation to mix tokens and the learned parameters correlate to the fixed input size, limiting the usability of dense prediction tasks.

To overcome the above limitation of fixed input sizes, more advanced MLP-based architectures [Lian *et al.*, 2022; Wang *et al.*, 2022a; Chen *et al.*, 2022; Guo *et al.*, 2022] were proposed to adapt arbitrary resolution last year. But they brought some new problems. Spatial shift [Lian *et al.*, 2022; Wang *et al.*, 2022a] operation is proposed to aggregate spatial information to make it feasible for arbitrary resolutions. But it only covers the local receptive field, which contradicts dense prediction tasks. CycleMLP [Chen *et al.*, 2022] is friendly to dense prediction, but sample points in a cyclical style may lose important visual cues and lead to bad results, especially in dense prediction tasks that require dealing with small objects. Hire-MLP [Guo *et al.*, 2022] captures global context by circularly shifting all tokens along spatial directions, but may damage the positional prior information.

Driven by these ideas, this paper explores how to design a vision MLP backbone not only to tackle arbitrary image sizes and scales but also to capture rich visual cues for various visual tasks. We propose Region-aware MLP (RaMLP), a vision MLP backbone for visual recognition and dense prediction. RaMLP mainly consists of a well-designed module, Region-aware Mixing (RaM), to capture local and global information in a region-aware manner. And it can adapt to arbitrary input sizes. First, inspired by recent researches [Diao *et al.*, 2022] that find a simple spatial pooling can achieve competitive results against the attention module in transformers, we use a learnable pooling, to better capture local visual cues that are essential to the results, especially in dense prediction tasks. Second, we propose Dilated Fully-Connection (DFC) to aggregate these local visual cues to obtain global context. Third, we add a Region-aware layer to further adjust the spatial features, which can capture visual cues more robustly.

Our RaMLP achieves the best accuracy in ImageNet-1K image classification (see Figure 1) compared to state-of-the-art ViT-based, MLP-based, and CNN-based models with fewer parameters and FLOPs. Moreover, compared with state-of-the-art MLP-based models, Wave-MLP, our improvements (0.3% accuracy on the tiny scale, 0.4% on the small scale, and 0.5% on the base scale) are significant. Also, compared with the well-known ViT-based model, Swin Transformer, our improvements are 0.6%-1.6% accuracy using less computation.

Moreover, our RaMLP can be easily transferred to downstream dense prediction tasks and achieve great results. According to experimental results, our RaMLP outperforms state-of-the-art ViT-based, MLP-based, and CNN-based backbones on dense prediction tasks, including MS-COCO object detection, MS-COCO instance segmentation, and ADE20K semantic segmentation. In particular, our RaMLP outperforms previous state-of-the-art MLP-based backbones by a large margin (around 1.5% Apb or 1.0% mIoU improvements). It demonstrates the proposed RaM is effective for MLPs in dense prediction tasks.

The experimental results demonstrate not only the effectiveness of our model but the great potential of MLPs in both image classification and dense prediction. We believe this paper will raise more attention to MLPs for vision.

Our contributions can be summarized below:

- We introduce a vision MLP architecture named Region-aware MLP, which employs a well-designed module, Region-aware Mixing to capture visual dependence in a coarse-to-fine manner. It can cope with various image sizes and be transferred to dense prediction tasks easily.
- Our Region-aware Mixing can adaptively determine aggregation weights according to spatial features, which can capture spatial visual cues more robustly and lead to more robust spatial feature extraction.
- Extensive experiments demonstrate that RaMLP outperforms the state-of-the-art CNNs, ViTs, and MLPs in various vision tasks, including image classification, object detection, instance segmentation, and semantic segmentation.

2 Related Work

CNN-based Architectures. After AlexNet [Krizhevsky *et al.*, 2012] won the 2012 ImageNet competition with an extremely great advantage, more and more CNN architectures were proposed. VGGNet [Simonyan and Zisserman, 2015] is a simple variant of Alexnet, by repeatedly stacking more convolutional layers. ResNet [He *et al.*, 2016a; He *et al.*, 2016b] explores the influence of depth. It even trains a 1001-layer network by an identity mapping branch. Inception models [Szegedy *et al.*, 2015; Ioffe and Szegedy, 2015; Szegedy *et al.*, 2016; Szegedy *et al.*, 2017] design a series of multi-branch architectures, to indicate the importance of multi-scale information. These works provide efficient structure, and their variants are widely used in the succeeding works. Recently, researchers introduced Transformers to visual recognition and proposed Vision Transformers, which superseded CNNs on many visual tasks. ConvNeXt [Liu *et al.*, 2022] discovers several key components to the performance and competes favorably with ViTs in terms of accuracy. However, ConvNeXt still inherits the weakness of CNN. The receptive field is far less than ViTs and MLPs, and sharing the same weights on spatial dimension also leads to a negative impact on extracting visual elements. Our RaMLP solves the above two problems at the same time.

Transformer-based Architectures. Due to the successful applications in natural language processing [Devlin *et al.*, 2019; Brown *et al.*, 2020], recent works, called Vision Transformers (ViTs) [Dosovitskiy *et al.*, 2021; Touvron *et al.*, 2021b], attempt to directly apply transformer to vision tasks such as image classification. They achieve comparable results with CNNs and even outperform using huge training data. However, directly applying self-attention to vision tasks leads to large computational costs, which is unacceptable for dense prediction tasks. Swin Transformer [Liu *et al.*, 2021] introduces pyramid structure and non-overlapping window partitions to ViTs. Thus it has linear computational complexity with respect to input image size. Recently, researchers

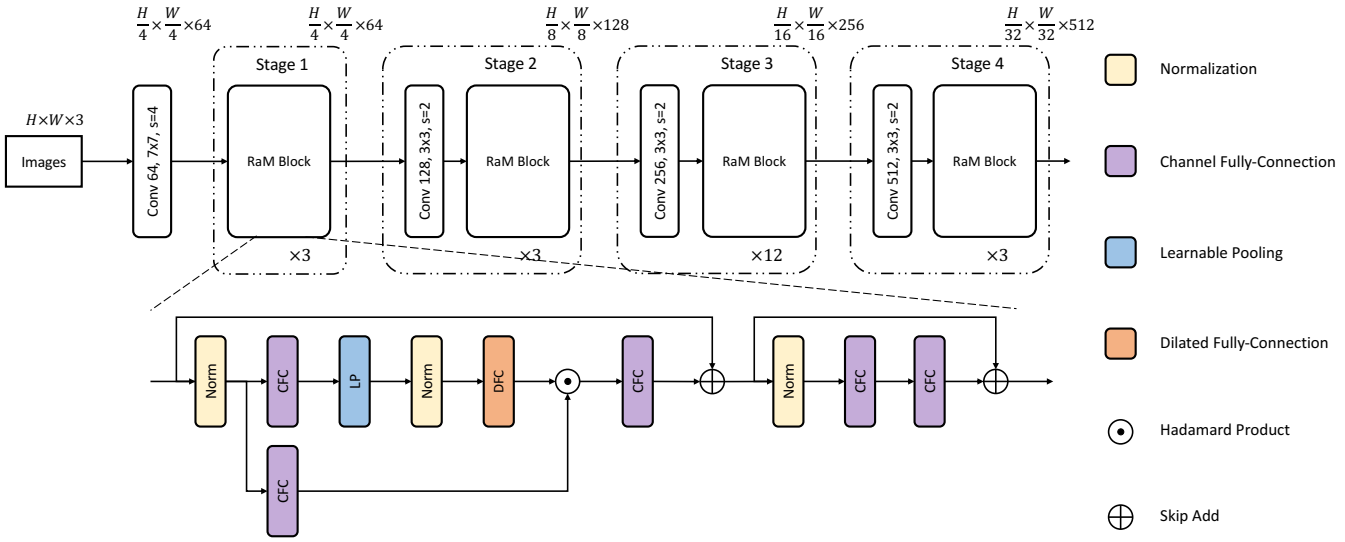

Figure 2: The overall architecture of our Tiny Region-aware MLP. It consists of several convolution layers for downsampling and some critical Region-aware Mixing blocks used to produce hierarchical representation.

also pointed out that ViTs and CNNs are complementary. CvT [Wu *et al.*, 2021a] and LocalViT [Li *et al.*, 2021] insert depthwise convolution into Multi-head Self-Attention or MLP module to enhance local context. CPVT [Chu *et al.*, 2021b] also adds an extra depthwise convolution, to generate conditional position encoding dynamically by the local neighborhood of the input tokens. However, compared with CNNs and MLPs, the components in ViTs are not friendly to most of the existing hardware, which limits their application.

MLP-based Architectures. MLP-Mixer [Tolstikhin *et al.*, 2021] and ResMLP [Touvron *et al.*, 2021a] are proposed almost at the same time, which shows that multi-layer perceptrons can also attain good accuracy/complexity trade-offs on ImageNet. To reduce the computational complexity but still capture long-range dependencies, ViP [Hou *et al.*, 2021] separately encodes the feature representations along the height and width dimensions and then aggregates the outputs in a mutually complementing manner. However, all these methods can only cope with fixed image size and are unfriendly to dense prediction tasks. Shift [Wang *et al.*, 2022a] and AS-MLP [Lian *et al.*, 2022] aggregate spatial information with spatial shift operation along spatial dimensions to make it flexible with various image sizes. CycleMLP [Chen *et al.*, 2022] is friendly to dense prediction, but sampling points in a cyclical style may lose important visual cues and lead to bad results, especially in dense prediction tasks that require dealing with small objects. Hire-MLP [Guo *et al.*, 2022] proposes the cross-region rearrangement to enable information communication between different regions by circularly shifting but may affect the positional prior. These models lack the ability to capture rich long-range dependencies, leading to unsatisfactory results for dense prediction in downstream tasks. Our RaMLP uses RaM to capture all visual dependencies in a coarse-to-fine manner and is seamlessly used for dense prediction.

3 Method

In this section, we first describe the overall architecture of RaMLP. Then we make a detailed introduction of the Region-aware Mixing (RaM) module, which is the key component of RaM block. Finally, we give brief configurations of the architecture variants.

3.1 Overall Architecture

An overview of the RaMLP architecture is presented in Figure 2, which illustrates the tiny version with $H \times W$ image. Followed by existing MLP-based architectures, we use a naive convolution layer for tokenizing input images and also three naive convolution layers for token merging across different RaM blocks.

The RaM blocks are the main components of our network, they are MLP-based architecture to enhance the representation of tokens before merging. We will introduce the details of RaMLP below.

3.2 Region-aware Mixing

The standard spatial FC used in MLP-Mixer [Tolstikhin *et al.*, 2021] and ResMLP [Touvron *et al.*, 2021a] computes all relations between tokens. The complexity is unacceptable, and FC weights are correlated to the number of tokens, which requires a fixed image scale and thus is infeasible for dense prediction. The spatial shift is a computation-free operation to overcome the above problems and thus is introduced by some recent MLP-based architectures [Lian *et al.*, 2022; Wang *et al.*, 2022a; Guo *et al.*, 2022]. But it cannot model long-range visual dependencies well, which is critical in dense prediction. CycleMLP [Chen *et al.*, 2022] is dense prediction friendly, but its cyclical sampling limits it to capture some visual cues, especially for small objects.

To overcome the above problems, we propose Region-aware Mixing (RaM) to capture visual dependence in a coarse-to-fine region-aware manner and adapt arbitrary input

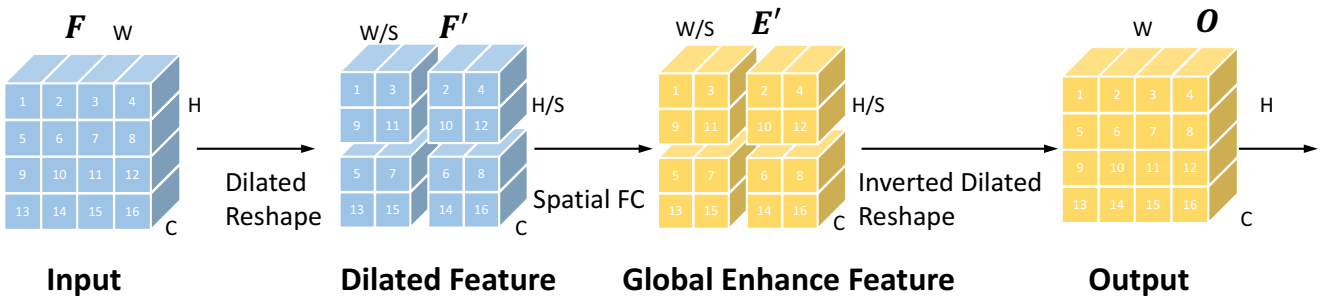

Figure 3: Illustration of Dilated Fully-Connection. DFC uses spatial Fully-Connection in every dilated feature, to model long-range visual dependencies in a sliding manner.

sizes. As illustrated in Figure 2, a RaM block consists of a Learnable Pooling (LP), a Dilated Fully-Connection (DFC), and five Channel Fully-Connection (CFC) layers. Three layernorm layers and two residual connections are applied for each block. The last layernorm layer, the last two CFC layers, and the last residual connection constitute a channel MLP module. Besides, same as Twins [Chu *et al.*, 2021a], we also introduce Conditional Positional Encoding (CPE), to handle the local positional information better. With these modules, RaM blocks are computed as:

$$t^l = x^{l-1} + \text{CPE}(x^{l-1}), \tag{1}$$
$$u^l = \text{LN}(t^l), \tag{2}$$
$$v^l = \text{CFC}(u^l), \tag{3}$$
$$w^l = \text{LP}(v^l), \tag{4}$$
$$y^l = \text{DFC}(\text{LN}(w^l)), \tag{5}$$
$$z^l = \text{CFC}(u^l), \tag{6}$$
$$s^l = t^l + \text{CFC}(y^l \cdot z^l), \tag{7}$$
$$x^l = s^l + \text{CFC}(\text{CFC}(\text{LN}(s^l))) \tag{8}$$

where $\text{LN}(\cdot)$ refers to layer normalization, and $t^l, u^l, v^l, w^l, y^l, z^l, s^l$ and x^l mean the outputs of these operations. The CPE is implemented as a simple depthwise convolution, which is widely used in previous works [Chu *et al.*, 2021a; Dong *et al.*, 2022] for its compatibility with the arbitrary size of the input.

Learnable Pooling. Inspired by recent researches [Diao *et al.*, 2022] that find a simple spatial pooling can achieve competitive results against the attention module in transformers, we also use a variant of pooling to better capture local visual cues that are essential to the results. For every spatial positional feature in spatial pooling, we assign learnable weight to better aggregate local visual cues. Actually, we could find it is very similar to a depthwise convolution. Thus, our Learnable Pooling is implemented as a simple depthwise convolution.

Dilated Fully-Connection. Dilated Fully-Connection (DFC) is a novel module to model long-range visual dependencies. As shown in Figure 3, we first reshape the input in dilated manner, to obtain dilated feature. It could be partitioned into several sparse global regions, and every region samples the point all over the input feature map. Specifically, given the input feature $F \in \mathbb{R}^{C\times H\times W}$, we partition it into $\frac{W}{S} \times \frac{H}{S}$ non-overlapping regions with a fixed region size of $S \times S$ in a dilated manner, to produce dilated feature $F^{'}$. Then, we perform spatial FC for every region, to obtain $\frac{W}{S} \times \frac{H}{S}$ global enhance feature $E^{'}$ with a fixed region size of $S \times S$. At last, we use an inverted dilated reshape to restore the position of features, to obtain the final output.

Region-aware Layer. Features after LP and DFC could capture all visual cues, but too much spatial information may introduce noises and easily lead to over-fitting. To solve the above issues, we add a Region-aware layer to further adjust the spatial importance of the features, which can capture visual cues more robustly. Specifically, we perform channel FC for input features and then do Hadamard Product between it and the output of DFC O. Thus, we adaptively determine aggregation over the whole spatial dimension and produce more robust regional features.

3.3 Architecture Variants

We build three models, called RaMLP-T (Tiny), RaMLP-S (Small), and RaMLP-B, to have the model size and computation complexity similar to Hire-MLP and Wave-MLP. We show the detailed configurations of all variants in Table 1.

4 Experiments

In this section, we first examine RaMLP by conducting experiments on ImageNet-1K [Deng *et al.*, 2009] image classification, and then for dense prediction tasks, including MS-COCO [Lin *et al.*, 2014] object detection, MS-COCO instance segmentation, and ADE20K [Zhou *et al.*, 2019] semantic segmentation.

4.1 ImageNet-1K Classification

Settings. We train our models on the ImageNet-1K [Deng *et al.*, 2009] dataset from scratch, which contains 1.2M training images and 50K validation images evenly spreading 1,000 categories. We report the top-1 accuracy on the validation set following the standard practice in this community. For fair comparisons, our training strategy is mostly adopted from CycleMLP, including RandAugment, Mixup, Cutmix, random erasing, and stochastic depth. AdamW and cosine learning rate schedules with the initial value of 1×10^{-3} are

Output stride	Layer	RaMLP-T	RaMLP-S	RaMLP-B
4	Patch Merging	$P_1 = 4$ $C_1 = 64$	$P_1 = 4$ $C_1 = 64$	$P_1 = 4$ $C_1 = 80$
	RaM block	$\begin{bmatrix} K_1 = 5 \\ S_1 = 8 \\ M_1 = 4 \\ E_1 = 3 \end{bmatrix} \times 3$	$\begin{bmatrix} K_1 = 5 \\ S_1 = 8 \\ M_1 = 4 \\ E_1 = 3 \end{bmatrix} \times 3$	$\begin{bmatrix} K_1 = 5 \\ S_1 = 8 \\ M_1 = 3 \\ E_1 = 3 \end{bmatrix} \times 3$
8	Patch Merging	$P_2 = 2$ $C_2 = 128$	$P_2 = 2$ $C_2 = 128$	$P_2 = 2$ $C_2 = 160$
	RaM block	$\begin{bmatrix} K_2 = 5 \\ S_2 = 4 \\ M_2 = 4 \\ E_2 = 3 \end{bmatrix} \times 3$	$\begin{bmatrix} K_2 = 5 \\ S_2 = 4 \\ M_2 = 4 \\ E_2 = 3 \end{bmatrix} \times 8$	$\begin{bmatrix} K_2 = 5 \\ S_2 = 4 \\ M_2 = 3 \\ E_2 = 3 \end{bmatrix} \times 8$
16	Patch Merging	$P_3 = 2$ $C_3 = 256$	$P_3 = 2$ $C_3 = 256$	$P_3 = 2$ $C_3 = 320$
	RaM block	$\begin{bmatrix} K_3 = 5 \\ S_3 = 2 \\ M_3 = 3 \\ E_3 = 2 \end{bmatrix} \times 12$	$\begin{bmatrix} K_3 = 5 \\ S_3 = 2 \\ M_3 = 3 \\ E_3 = 2 \end{bmatrix} \times 26$	$\begin{bmatrix} K_3 = 5 \\ S_3 = 2 \\ M_3 = 2 \\ E_3 = 2 \end{bmatrix} \times 26$
32	Patch Merging	$P_4 = 2$ $C_4 = 512$	$P_4 = 2$ $C_4 = 512$	$P_4 = 2$ $C_4 = 640$
	RaM block	$\begin{bmatrix} K_4 = 5 \\ S_4 = 1 \\ M_4 = 3 \\ E_4 = 2 \end{bmatrix} \times 3$	$\begin{bmatrix} K_4 = 5 \\ S_4 = 1 \\ M_4 = 3 \\ E_4 = 2 \end{bmatrix} \times 3$	$\begin{bmatrix} K_4 = 5 \\ S_4 = 1 \\ M_4 = 2 \\ E_4 = 2 \end{bmatrix} \times 3$
-	Params	25M	38M	58M
-	FLOPs	4.2G	7.8G	12.0G

Table 1: Overall architecture of RaMLP with three different levels of complexities. As shown in Section 3.3, P_l denotes the spatial reduction factor, C_l denotes the channel number, K_l denotes the kernel size for the LP, S_l denotes the region size for the DFC, E_l denotes the expansion ratio for channel FC in RaM, M_l denotes the expansion ratio for channel FC in channel MLP. FLOPs are evaluated on 224×224 resolution.

adopted. All models are trained for 300 epochs with a 20-epoch warm-up on Nvidia 3090 GPUs with a batch size of 512.

Comparison with MLP-based Models. We compare our RaMLP with MLP-based models proposed in recent two years, and we show the results in Table 2. First, we get a breakthrough in image classification. Hire-MLP-B, the previous state-of-the-art model, achieves 83.8% accuracy with 13.4G FLOPs and 96M parameters. In comparison, our RaMLP achieves the same accuracy using much less computation (7.8G FLOPs) and much fewer parameters (38M). Second, our RaMLP achieves the best results in three different computation scales (0-4G FLOPs, 4-8G FLOPs, and ¿8G FLOPs), demonstrating that our RaMLP performs well on different computation resources. Besides, Wave-MLP is the most related model designed for dense prediction tasks. Our RaMLP surpasses it by 0.2% accuracy with only 76% FLOPs (seeing the results of Wave-MLP-B, and our RaMLP-S). It demonstrates the effectiveness of the proposed modules to capture visual dependencies in a coarse-to-fine manner with region-aware modeling.

Comparison with SOTA Models. We compare our RaMLP with state-of-the-art models, including CNN-based models, ViT-based models, and MLP-based models, and we show the results in Table 3. First, it is encouraging to us that our RaMLP surpasses SOTAs in all four computation scales (4-6GFLOPs, 6-10GFLOPs, and 10GFLOPs). It demonstrates the great potential of MLP-based models, and it is promising to do more research on MLPs. Second, our RaMLP achieves more improvements in accuracy on a tiny scale, which means our model is suitable for low computation capability scenarios. Moreover, our method gets better results in accuracy/FLOPs trade-off compared with the state-of-the-art transformer-based models across different computation scales. It demonstrates that well-designed MLP modules may be more suitable than self-attention modules in computer vision tasks. In this experiment, we demonstrated the superiority of MLP-based models against CNN-based models and ViT-based models in image classification, the most common computer vision task. And thus, we think our research made a solid contribution to MLP research and can attract many followers to explore MLPs and bring more exciting results.

4.2 Ablation Study

In this section, we utilize RaMLP-T to verify the effectiveness of the proposed components by conducting extensive ablation studies.

Study on Region Size. We evaluate the effectiveness of adjusting the region size in Table 4 and find that increasing the region size could improve the performance. Small region size decreases the density of the sampling point and increases the loss of spatial information. Too small a region count even leads to non-convergence.

Study on Effectiveness of Different Components. As shown in Table 5, we set a RaMLP without RaM as a base-

Models	Top1	FLOPs	Params	Throughput
EAMLP-14	78.9	-	30M	771
EAMLP-19	79.4	-	55M	464
ResMLP-S12	76.6	3.0G	15M	1415
ResMLP-S24	79.4	6.0G	30M	715
ResMLP-B24	81.0	23.0G	116M	231
RepMLPNet-T†	77.5	4.2G	59M	1374
RepMLPNet-B†	81.0	9.6G	97M	708
RepMLPNet-L†	81.8	11.5G	118M	588
ViP-S/7	81.5	6.9G	25M	719
ViP-M/7	82.7	16.3G	55M	418
ViP-L/7	83.2	24.4G	88M	298
CycleMLP-T	81.3	4.4G	28M	611
CycleMLP-S	82.9	8.5G	50M	360
CycleMLP-B	83.4	15.2G	88M	216
AS-MLP-T	81.3	4.4G	28M	864
AS-MLP-S	83.1	8.5G	50M	478
AS-MLP-B	83.3	15.2G	88M	312
Shift-T	81.7	4.4G	28M	792
Shift-S	82.8	8.5G	50M	430
Shift-B	83.3	15.2G	88M	308
Hire-MLP-S	82.1	4.2G	33M	808
Hire-MLP-B	83.2	8.1G	58M	441
Hire-MLP-L	83.8	13.4G	96M	290
Wave-MLP-S	82.6	4.5G	30M	720
Wave-MLP-T	83.4	7.9G	44M	413
Wave-MLP-B	83.6	10.2G	63M	341
RaMLP-T	82.9	4.2G	25M	759
RaMLP-S	83.8	7.8G	38M	441
RaMLP-B	84.1	12.0G	58M	333

Table 2: Comparison with MLP-based models on ImageNet-1K image classification. All models are trained with the input resolution of 224×224, except † with 256×256.

line, then, we add LP, DFC, and Ra layer to verify the effectiveness. All these components have obvious effects.

4.3 Object Detection and Instance Segmentation

Settings. We conduct object detection experiments with RetinaNet [Lin *et al.*, 2017], and instance segmentation experiments with Mask R-CNN [He *et al.*, 2017] on COCO [Lin *et al.*, 2014] dataset by following the experimental settings of CycleMLP [Chen *et al.*, 2022].

Results on Object Detection. Object detection is a typical dense prediction task. We separate the models into three scales according to the number of parameters and show the results in Table 6. First, our RaMLP achieves the best results across three scales using nearly the least parameters, which demonstrates the effectiveness and efficiency of our RaMLP in dense prediction tasks. And interestingly, our improvement in the first scale is the hugest, which demonstrates that our model design is feasible for low computation resource scenarios. It is worth noting that our RaMLP outperforms the ResNet-50 and ResNet-101, the two most widely used CNN-based backbones, by a large margin (around 7% Apb) with fewer parameters. Moreover, our model outperforms previous state-of-the-art MLPs by a large margin. Compared with Hire-MLP, we get +1.9%, +1.2%, and +1.5% accuracy, respectively, using a similar number of parameters. It demonstrates the effectiveness of our proposed RaP and RaDFC for dense prediction.

Models	Arch.	Top1	FLOPs	Params
ResT-B	ViT	81.6	4.3G	30M
CvT-13	Hybrid	81.6	4.5G	20M
Swin-T	ViT	81.3	4.5G	29M
Focal-T	ViT	82.2	4.9G	29M
TNT-S	ViT	81.3	5.2G	24M
GFNet-H-S	FFT	81.5	4.5G	32M
PoolFormer-S36	CNN	81.4	5.2G	31M
TNT-S	ViT	81.5	5.2G	24M
I-D-DW-Conv.-T	CNN	81.8	4.4G	22M
ConvNeXt-T	CNN	82.1	4.5G	29M
DAT-T	ViT	82.0	4.6G	29M
RaMLP-T	MLP	82.9	4.2G	25M
CvT-21	Hybrid	82.5	7.1G	32M
BoT-S1-59	Hybrid	81.7	7.3G	34M
GFNet-H-B	FFT	82.9	8.4G	54M
Swin-S	ViT	83.0	8.7G	50M
Focal-S	ViT	83.6	9.4G	51M
ConvNeXt-S	CNN	83.1	8.7G	50M
PoolFormer-M36	CNN	82.1	9.1G	56M
PVT-Large	ViT	81.7	9.8G	61M
DAT-S	ViT	83.7	9.0G	50M
RaMLP-S	MLP	83.8	7.8G	38M
PoolFormer-M48	CNN	82.5	11.9G	73M
T2T-ViT-24	ViT	82.3	13.8G	64M
TNT-B	ViT	82.8	14.1G	66M
I-D-DW-Conv.-B	CNN	83.4	14.3G	80M
Swin-B	ViT	83.5	15.4G	88M
Focal-B	ViT	84.0	16.4G	90M
ConvNeXt-B	CNN	83.8	15.4G	89M
NViT-B	ViT	83.1	17.6G	86M
DAT-S	ViT	84.0	15.8G	88M
RaMLP-B	MLP	84.1	12.0G	58M

Table 3: Comparison with SOTA models on ImageNet-1K image classification.

S 1	S 2	S 3	S 4	Top1	FLOPs	Params
1	1	1	1	NaN	3.9G	24M
2	1	1	1	NaN	4.0G	24M
4	2	1	1	82.7	4.1G	25M
8	4	2	1	82.9	4.2G	25M

Table 4: The impacts of the region size. S means stage.

Results on Instance Segmentation. Instance segmentation is a more challenging dense prediction task against object detection. Following the evaluation in object detection, we separate the models into three scales and show the results in Table 7. First, compared with object detection, our improvements to the state-of-the-art is more obvious (more than 1% AP^b), which demonstrates the effectiveness of our RaMLP in dense prediction tasks. And the dense prediction task is more challenging, more improvements RaMLP can achieve. Second, other phenomena in object detection also occur, which demonstrates the generality of RaMLP to various dense predictions.

Settings. Following the PVT [Wang *et al.*, 2021], we evaluate the potential of RaMLP on the challenging semantic seg-

LP	DFC	Ra	Top1	FLOPs	Params
-	-	-	80.9	3.3G	21M
✓	-	-	81.8	3.4G	21M
-	✓	-	81.6	3.5G	21M
-	-	✓	82.0	3.8G	24M
✓	✓	-	82.3	3.6G	21M
✓	✓	✓	82.9	4.2G	25M

Table 5: The impacts of the components.

Models	Arch.	Params	AP^b	AP^b_{50}	AP^b_{75}
ResNet50	CNN	38	36.3	55.3	38.6
Pool-S24	CNN	31	38.9	59.7	41.3
PVT-S	ViT	34	40.4	61.3	43.0
Swin-T	ViT	29	41.5	62.1	44.2
CycleMLP-B2	MLP	37	40.9	61.8	43.4
Hire-MLP-S	MLP	43	41.7	-	-
RaMLP-T	MLP	35	43.6	64.9	46.8
ResNet101	CNN	57	38.5	57.8	41.2
Pool-S36	CNN	41	39.5	60.5	41.8
PVT-M	ViT	54	41.9	63.1	44.3
Swin-S	ViT	60	44.5	65.7	47.5
CycleMLP-B3	MLP	48	42.5	63.2	45.3
Hire-MLP-B	MLP	68	44.3	-	-
RaMLP-S	MLP	49	45.5	66.7	48.5
PVT-L	ViT	71	42.6	63.7	45.4
Swin-B	ViT	98	44.7	65.9	47.8
CycleMLP-B4	MLP	62	43.2	63.9	46.2
Hire-MLP-L	MLP	106	44.9	-	-
RaMLP-B	MLP	70	46.4	67.7	49.7

Table 6: Object detection on COCO val2017 with RetinaNet.

mentation task on ADE20K [Zhou *et al.*, 2019], which contains 20K training and 2K validation images. We adopt Semantic FPN [Kirillov *et al.*, 2019], with RaMLP pretrained on ImageNet-1K [Deng *et al.*, 2009] as the backbone. We train 40K iterations with a batch size of 32.

Results. Semantic segmentation is also one of the most common dense prediction tasks. We separate the models into three scales according to FLOPs and show the results in Table 8. First, impressively, our RaMLP outperforms previous SOTAs by a large margin (0.9%, 1.3%, and 1.5% improvements on three scales, respectively). It is interesting that Hire-MLP, the previous state-of-the-art MLP-based model, does not show significant superiority against transformer-based models, but our RaMLP does. Hire-MLP uses hierarchical rearrangement to capture spatial information but may lose important visual cues in semantic segmentation, while RaMLP can capture rich visual cues for various visual tasks. Second, our RaMLP achieves the best results using the least computation and parameters in the second and third scales.

5 Conclusion

We introduce a new MLP-based architecture named Region-aware MLP (RaMLP) with a well-designed module, Region-aware Mixing (RaM), to capture visual dependence in a coarse-to-fine region-aware manner. It can adaptively determine aggregation weights according to regions and inputs, to extract regional features more robustly. It also can cope with various image sizes and be transferred to dense prediction tasks easily. The results on image classification, object detection, instance segmentation, and semantic segmentation, show that our RaMLP outperforms the SOTAs.

Models	Arch.	Params	AP^b	AP^m
ResNet50	CNN	44	38.0	34.4
PVT-S	ViT	44	40.4	37.8
Swin-T	ViT	48	43.7	39.8
CycleMLP-B2	MLP	47	42.1	38.9
Hire-MLP-S	MLP	53	42.8	39.3
RaMLP-T	MLP	45	44.8	41.0
ResNet101	CNN	63	40.4	36.4
PVT-M	ViT	64	42.0	39.0
Swin-S	ViT	69	44.8	40.9
CycleMLP-B3	MLP	58	43.4	39.5
Hire-MLP-B	MLP	78	45.2	41.0
RaMLP-S	MLP	61	46.9	42.5
ResNeXt101-64x4d	CNN	102	42.8	38.4
PVT-L	ViT	81	42.9	39.5
Swin-B	ViT	107	45.5	42.1
CycleMLP-B4	MLP	72	44.1	40.2
Hire-MLP-L	MLP	115	45.9	41.7
RaMLP-B	MLP	81	47.4	42.8

Table 7: Instance segmentation on COCO val2017 with Mask R-CNN.

Models	Arch.	Top1	FLOPs	Params
ResNet50	CNN	36.7	46G	29M
PVT-S	ViT	39.8	45G	28M
Swin-T	ViT	41.5	46G	32M
CycleMLP-B2	MLP	43.4	42G	31M
Hire-MLP-S	MLP	44.3	44G	37M
RaMLP-T	MLP	46.1	42G	29M
ResNet101	CNN	38.8	65G	48M
PVT-M	ViT	41.6	61G	48M
Swin-S	ViT	45.2	70G	53M
CycleMLP-B3	MLP	44.3	58G	42M
Hire-MLP-B	MLP	46.2	64G	62M
RaMLP-S	MLP	47.5	63G	44M
ResNeXt101	CNN	38.8	104G	86M
PVT-L	ViT	41.6	80G	65M
Swin-B	ViT	44.9	107G	91M
CycleMLP-B4	MLP	45.1	75G	56M
Hire-MLP-L	MLP	46.6	92G	99M
RaMLP-B	MLP	48.1	89G	63M

Table 8: Semantic segmentation on ADE20K Val with Semantic FPN. FLOPs are evaluated on 512×512 resolution. All backbones are pretrained on ImageNet-1K.

Acknowledgments

Shenqi Lai and Xi Du contribute equally. Kaipeng Zhang is the corresponding author. This work is partially supported by the National Key R&D Program of China(NO.2022ZD0160100) and in part by Shanghai Committee of Science and Technology (Grant No. 21DZ1100100).

References

[Brown *et al.*, 2020] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020.

[Chen *et al.*, 2022] Shoufa Chen, Enze Xie, Chongjian Ge, Runjian Chen, Ding Liang, and Ping Luo. Cyclemlp: A mlp-like architecture for dense prediction. In *ICLR*, 2022.

[Chu *et al.*, 2021a] 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. In *NeurIPS*, 2021.

[Chu *et al.*, 2021b] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv:2102.10882*, 2021.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[Diao *et al.*, 2022] Qishuai Diao, Yi Jiang, Bin Wen, Jia Sun, and Zehuan Yuan. Metaformer: A unified meta framework for fine-grained recognition. *arXiv:2203.02751*, 2022.

[Dong *et al.*, 2022] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. Cswin transformer: A general vision transformer backbone with cross-shaped windows. In *CVPR*, 2022.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[Guo *et al.*, 2021] Meng-Hao Guo, Zheng-Ning Liu, Tai-Jiang Mu, and Shi-Min Hu. Beyond self-attention: External attention using two linear layers for visual tasks. *arXiv:2105.02358*, 2021.

[Guo *et al.*, 2022] Jianyuan Guo, Yehui Tang, Kai Han, Xinghao Chen, Han Wu, Chao Xu, Chang Xu, and Yunhe Wang. Hire-mlp: Vision MLP via hierarchical rearrangement. In *CVPR*, 2022.

[He *et al.*, 2016a] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[He *et al.*, 2016b] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.

[He *et al.*, 2017] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, 2017.

[Hou *et al.*, 2021] Qibin Hou, Zihang Jiang, Li Yuan, Ming-Ming Cheng, Shuicheng Yan, and Jiashi Feng. Vision permutator: A permutable mlp-like architecture for visual recognition. *arXiv:2106.12368*, 2021.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[Kirillov *et al.*, 2019] Alexander Kirillov, Ross B. Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

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

[Lian *et al.*, 2022] Dongze Lian, Zehao Yu, Xing Sun, and Shenghua Gao. AS-MLP: an axial shifted MLP architecture for vision. In *ICLR*, 2022.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014.

[Lin *et al.*, 2017] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

[Liu *et al.*, 2022] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *arXiv:2201.03545*, 2022.

[Simonyan and Zisserman, 2015] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[Szegedy *et al.*, 2015] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[Szegedy *et al.*, 2016] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.

[Szegedy *et al.*, 2017] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. Inception-v4,

inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[Tolstikhin *et al.*, 2021] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. *arXiv:2105.01601*, 2021.

[Touvron *et al.*, 2021a] Hugo Touvron, Piotr Bojanowski, Mathilde Caron, Matthieu Cord, Alaaeldin El-Nouby, Edouard Grave, Armand Joulin, Gabriel Synnaeve, Jakob Verbeek, and Hervé Jégou. Resmlp: Feedforward networks for image classification with data-efficient training. *arXiv:2105.03404*, 2021.

[Touvron *et al.*, 2021b] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.

[Wang *et al.*, 2021] 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 *ICCV*, 2021.

[Wang *et al.*, 2022a] Guangting Wang, Yucheng Zhao, Chuanxin Tang, Chong Luo, and Wenjun Zeng. When shift operation meets vision transformer: An extremely simple alternative to attention mechanism. In *AAAI*, 2022.

[Wang *et al.*, 2022b] Ziyu Wang, Wenhao Jiang, Yiming Zhu, Li Yuan, Yibing Song, and Wei Liu. Dynamixer: A vision MLP architecture with dynamic mixing. In *ICML*, 2022.

[Wu *et al.*, 2021a] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *ICCV*, 2021.

[Wu *et al.*, 2021b] Yu-Huan Wu, Yun Liu, Xin Zhan, and Ming-Ming Cheng. P2T: pyramid pooling transformer for scene understanding. *arXiv:2106.12011*, 2021.

[Yang *et al.*, 2021] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal attention for long-range interactions in vision transformers. In *NeurIPS*, 2021.

[Yuan *et al.*, 2021] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis E.H. Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *ICCV*, 2021.

[Zhou *et al.*, 2019] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *IJCV*, 2019.