南京大學
NANJING UNIVERSITY

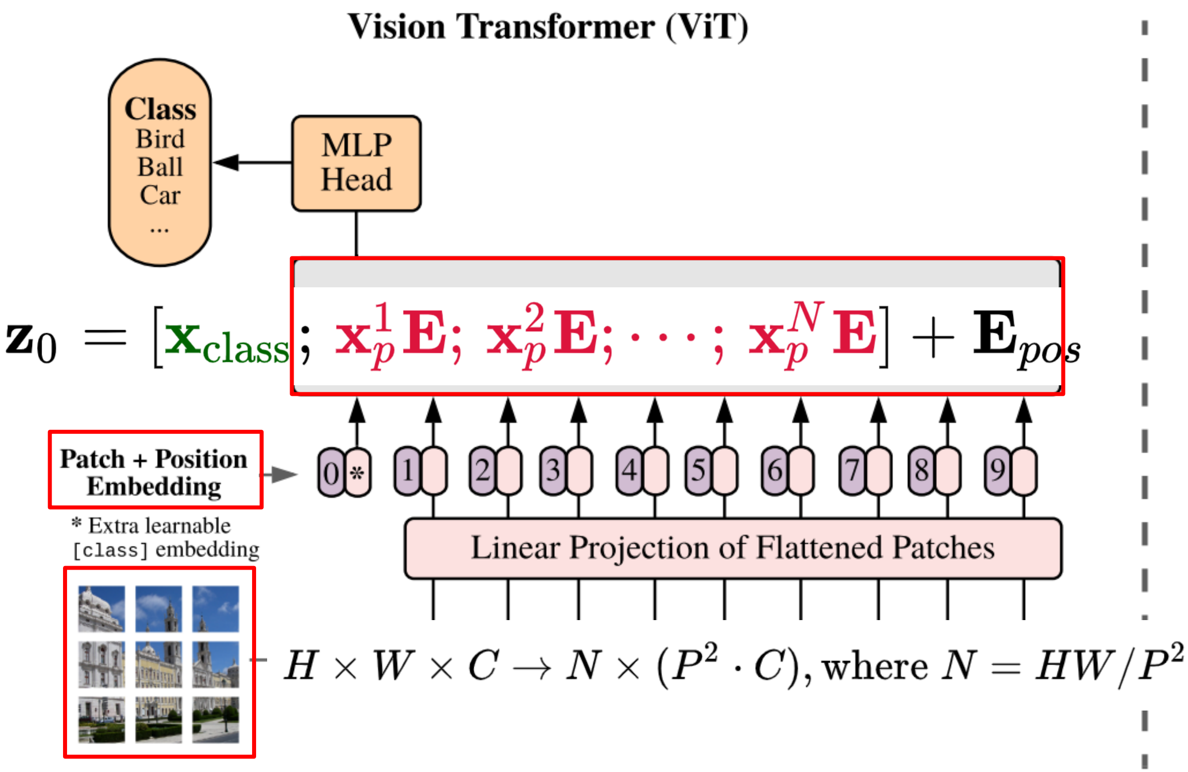
ResT: An Efficient Transformer for Visual Recognition

Presenter：Qing-Long Zhang

Content

Introduction

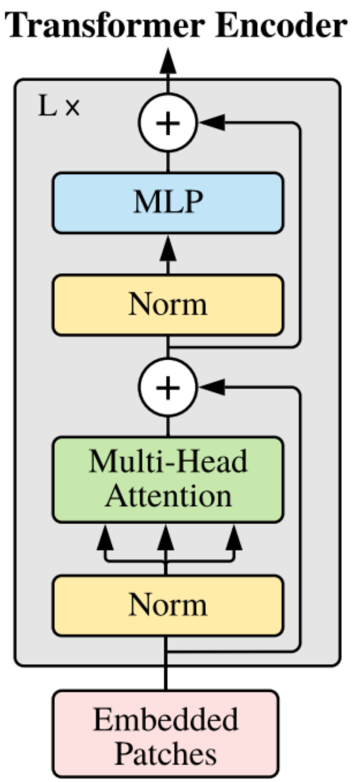
- **ViT**



Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." (ICLR2021)

Introduction

- **ViT**



Let $x \in \mathbb{R}^{n\times d_m}$ be the input token, the output of each block

$$\mathrm{y} = \mathrm{x}' + \mathrm{FFN}(\mathrm{LN}(\mathrm{x}')), \text{ and } \mathrm{x}' = \mathrm{x} + \mathrm{MSA}(\mathrm{LN}(\mathrm{x})) \quad (1)$$

In MSA, x is split into k heads, each with size $n \times d_k$, then the results of one head can be represented as

$$\mathrm{SA}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{Softmax}(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_k}})\mathbf{V} \quad (2)$$

FFN contains 2 linear layers with a non-linearity activation

$$\mathrm{FFN}(\mathrm{x}) = \sigma(\mathrm{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \quad (3)$$

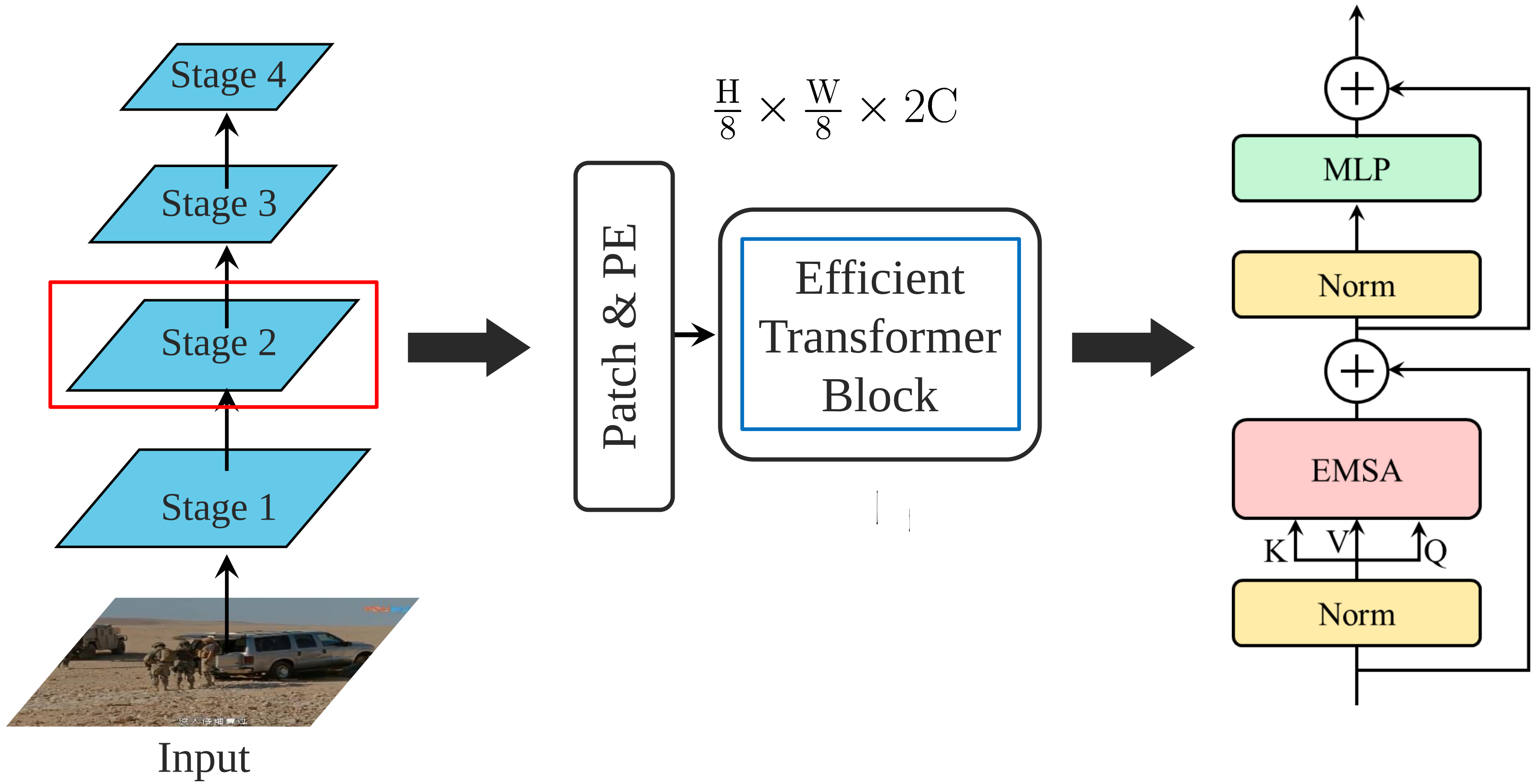
Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." (ICLR2021)

Introduction

- **Shortcomings of ViT**
  - Non-Overlapping Patch Embedding is difficult to extract the low-level features which form some fundamental structures in images.
  - Input token and PE are all of a fixed scale, unsuitable for dense prediction.
  - Computation of MSA is $\mathcal{O}(2d_m n^2 + 4d_m^2 n)$, causing vast overheads for training and inference.
  - Each head in MSA is responsible for only a subset of embedding dims d_k , which may impair the performance of the network, particularly when the tokens embedding dimension (for each head) is short.

ResT

- **Pipeline**



ResT

- **Patch Embedding**
  - The patch embedding module creates a multi-scale pyramid of features by hierarchically expanding the channel capacity while reducing the spatial resolution with overlapping convolution operations.
  - At the beginning of each stage, a standard Conv-3 with stride 2 and padding 1 is adopted to down-sample the spatial dimension by 4x and increase the channel dimension by 2x.
  - The first Patch embedding module is applied with three consecutive Conv-3 with stride 2, 1, 2.

ResT

- **Positional Encoding**

Let $x \in \mathbb{R}^{n \times d_m}$ be the input token, $\theta \in \mathbb{R}^{n \times d_m}$ be learnable parameters, PE in ViT can be represented as

$$\hat{x} = x + \theta$$

If θ is related to x, then PE can be represented as

$$\hat{x} = x + \mathrm{GL}(x)$$

PE can be further constructed as spatial attention

$$\hat{x} = x * \mathrm{SpatialAttention}(x)$$

ResT

- **Positional Encoding**

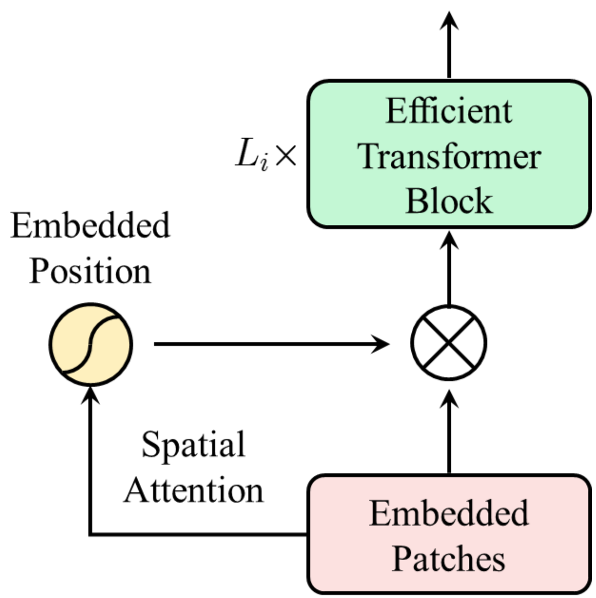
Table 7: Comparison of various position encoding (PE) strategies on ResT-Lite.

Encoding	Top-1 (%)	Top-5 (%)
w/o position	71.54	89.82
+ LE	71.98	90.32
+ GL	72.04	90.41
+ PA	72.88	90.62

ResT

- **Patch Embedding & Positional Encoding**

Since the input token in each stage is obtained by a convolutional operation, we can embed PE into the patch embedding module.



ResT

- **EMSA**

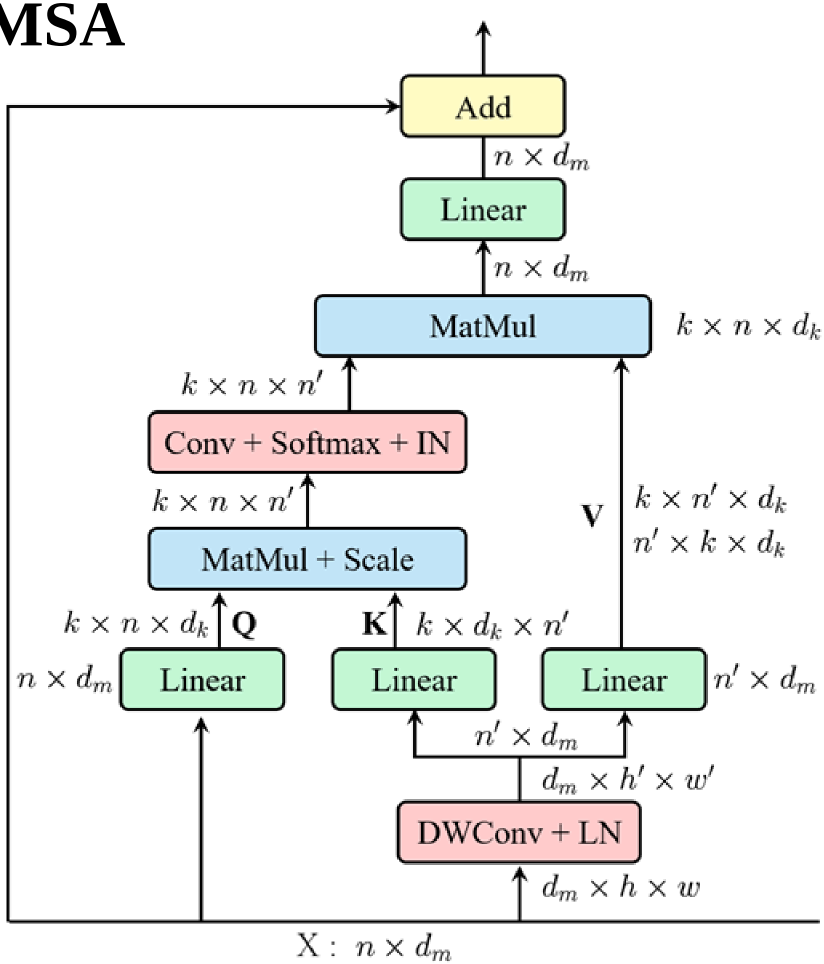
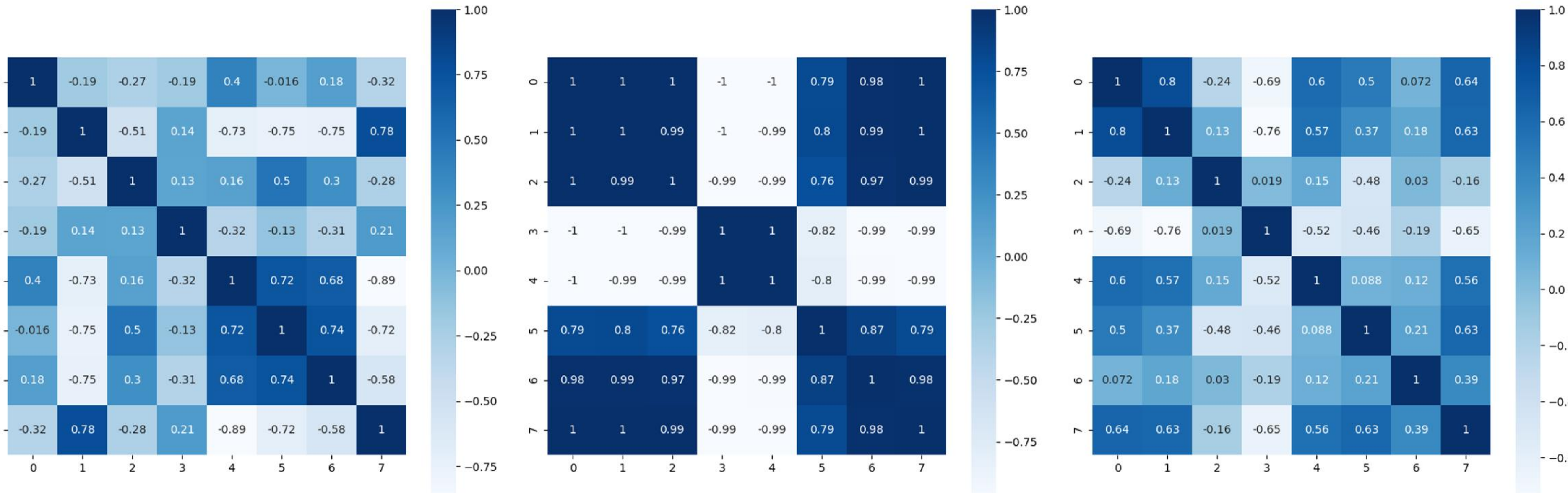


Table 6: Comparison of different reduction strategies of EMSA on ResT-Lite. Results show that Average Pooling can be an alternative to Depthwise Conv2d to make a trade-off.

Reduction	Top-1 (%)	Top-5 (%)
DWConv	72.88	90.62
Avg Pooling	72.64	90.41
Max Pooling	72.20	89.97

- **EMSA**



(a) Before Conv (b) After Conv (c) After IN

Figure : Attention map visualization of the last blocks of stage 4 of the ResT-Lite.

- **EMSA**

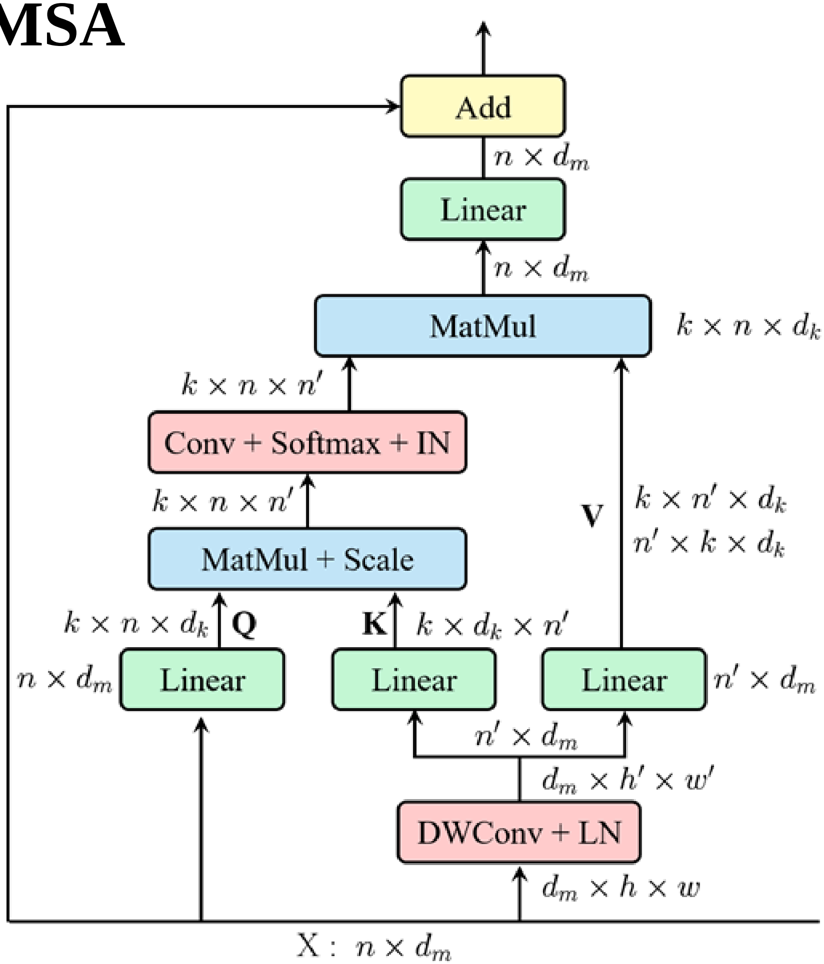


Table 7: Ablation study results on the important design elements of EMSA on ResT-Lite, including the 1×1 convolution operation and Instance Normalization in Eq. 4.

Methods	Top-1 (%)	Top-5 (%)
origin	72.88	90.62
w/o IN	71.98	90.32
w/o Conv-1&IN	71.72	89.93

ResT

- **EMSA vs. MSA**

EMSA Computation：

$$\mathcal{O}\left(\frac{2d_m n^2}{s^2} + 2d_m^2 n\left(1 + \frac{1}{s^2}\right)\right)$$

MSA Compuation：

$$\mathcal{O}(2d_m n^2 + 4d_m^2 n)$$

Table 8: Comparison of MSA and EMSA.

Model	#Params (M)	FLOPs (G)	Throughput	Top-1 (%)	Top-5 (%)
MSA	10.48	1.6	512	72.68	90.46
EMSA	10.49	1.4	1246	72.88	90.62

ResT

- **Architecture of ResT**

Name	Output	Lite	Small	Base	Large
stem	56×56	patch_embed: Conv-3_C/2_2, Conv-3_C/2_1, Conv-3_C_2,PA			
stage1	56×56	$\begin{bmatrix} \text{EMSA\_1\_8} \\ \text{MLP\_64} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_1\_8} \\ \text{MLP\_64} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_1\_8} \\ \text{MLP\_96} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_1\_8} \\ \text{MLP\_96} \end{bmatrix} \times 2$
stage2	28×28	patch_embed: Conv-3_2C_2, PA			
		$\begin{bmatrix} \text{EMSA\_2\_4} \\ \text{MLP\_128} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_2\_4} \\ \text{MLP\_128} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_2\_4} \\ \text{MLP\_192} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_2\_4} \\ \text{MLP\_192} \end{bmatrix} \times 2$
stage3	14×14	patch_embed: Conv-3_4C_2, PA			
		$\begin{bmatrix} \text{EMSA\_4\_2} \\ \text{MLP\_256} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_4\_2} \\ \text{MLP\_256} \end{bmatrix} \times 6$	$\begin{bmatrix} \text{EMSA\_4\_2} \\ \text{MLP\_384} \end{bmatrix} \times 6$	$\begin{bmatrix} \text{EMSA\_4\_2} \\ \text{MLP\_384} \end{bmatrix} \times 18$
stage4	7×7	patch_embed: Conv-3_8C_2, PA			
		$\begin{bmatrix} \text{EMSA\_8\_1} \\ \text{MLP\_512} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_8\_1} \\ \text{MLP\_512} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_8\_1} \\ \text{MLP\_768} \end{bmatrix} \times 2$	$\begin{bmatrix} \text{EMSA\_8\_1} \\ \text{MLP\_768} \end{bmatrix} \times 2$
Classifier	1×1	average pool, 1000d fully-connected			
GFLOPs		1.4	1.94	4.26	7.91

ResT

Model	#Params (M)	FLOPs (G)	Throughput	Top-1 (%)	Top-5 (%)
ConvNet					
ResNet-18 [10]	11.7	1.8	1852	69.7	89.1
ResNet-50 [10]	25.6	4.1	871	79.0	94.4
ResNet-101 [10]	44.7	7.9	635	80.3	95.2
RegNetY-4G [21]	20.6	4.0	1156	79.4	94.7
RegNetY-8G [21]	39.2	8.0	591	79.9	94.9
RegNetY-16G [21]	83.6	15.9	334	80.4	95.1
Transformer					
DeiT-S [25]	22.1	4.6	940	79.8	94.9
DeiT-B [25]	86.6	17.6	292	81.8	95.6
PVT-T [28]	13.2	1.9	1038	75.1	92.4
PVT-S [28]	24.5	3.7	820	79.8	94.9
PVT-M [28]	44.2	6.4	526	81.2	95.6
PVT-L [28]	61.4	9.5	367	81.7	95.9
Swin-T [18]	28.29	4.5	755	81.3	95.5
Swin-S [18]	49.61	8.7	437	83.3	96.2
Swin-B [18]	87.77	15.4	278	83.5	96.5
MViT-B-16 [8]	37.0	7.8	-	83.0	
ResT-Lite (Ours)	10.49	1.4	1246	**77.2** (↑ 7.5)	**93.7** (↑ 4.6)
ResT-Small (Ours)	13.66	1.9	1043	**79.6** (↑ 9.9)	**94.9** (↑ 5.8)
ResT-Base (Ours)	30.28	4.3	673	**81.6** (↑ 2.6)	**95.7** (↑ 1.3)
ResT-Large (Ours)	51.63	7.9	429	**83.6** (↑ 3.3)	**96.3** (↑ 1.1)

ResT

- **Object Detection on MS COCO**

Table 3: Object detection performance on the COCO val2017 split using the RetinaNet framework.

Backbones	AP50:95	AP50	AP75	APs	APm	APl	Param (M)
R18 [10]	31.8	49.6	33.6	16.3	34.3	43.2	21.3
PVT-T [28]	36.7	56.9	38.9	22.6	38.8	50.0	23.0
ResT-Small(Ours)	**40.3**	61.3	42.7	25.7	43.7	51.2	23.4
R50 [10]	37.4	56.7	40.3	23.1	41.6	48.3	37.9
PVT-S [28]	40.4	61.3	43.0	25.0	42.9	55.7	34.2
Swin-T [18]	41.5	62.1	44.1	27.0	44.2	53.2	38.5
ResT-Base (Ours)	**42.0**	63.2	44.8	29.1	45.3	53.3	40.5
R101 [10]	38.5	57.8	41.2	21.4	42.6	51.1	56.9
PVT-M [28]	41.9	63.1	44.3	25.0	44.9	57.6	53.9
Swin-S [18]	44.5	65.7	47.5	27.4	48.0	59.9	59.8
ResT-Large (Ours)	**44.8**	66.1	48.0	28.3	48.7	60.3	61.8

Conclusion

- we proposed ResT, an efficient multi-scale vision Transformer, which produces hierarchical feature representations for dense prediction.
- We build a EMSA, which compresses the memory by a simple depth-wise convolution, and models the interaction across the attention-heads dimension while keeping the diversity ability of multi-heads
- Position encoding is constructed as spatial attention, which is more flexible and can tackle with input images of arbitrary size without interpolation or fine-tune.
- We design an effective stem module, which consists of a stack of overlapping convolution operations with stride on the token map.

Thank you!