EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications

Muhammad Maaz[1,*] Abdelrahman Shaker[1,*] Hisham Cholakkal[1] Salman Khan[1,2]
Syed Waqas Zamir[3] Rao Muhammad Anwer[1] Fahad Shahbaz Khan[1]
[1]Mohamed Bin Zayed university of Artificial Intelligence [2]Australian National University
[3]Inception Institute of Artificial Intelligence

试图解决的问题：

背景：

In the pursuit of achieving ever-increasing accuracy, large and complex neural networks are usually developed. Such models demand high computational resources and therefore cannot be deployed on edge devices.

工作：

结合CNN和transformer提出了一个更加轻量化的模型，在参数量大幅减少的情况下，提升推理速度，并依然能在分类、识别、语义分割任务中保持不错的正确率和泛化性。

相关工作:

1 designing efficient versions of convolutions:

- MobileNet: depth-wise separable convolutions
- ShuffleNet: channel shuffling and low-cost group convolutions

2 hardware-aware neural architecture search (NAS)

- 体系结构相关
- 神经网络加速器

CNN的两大缺点:

1. It has local receptive field and thereby unable to model global context.
2. The learned weights are stationary at inference times, making CNNs inflexible to adapt to the input content.

解决方案: transformer?

相关工作:

论文《An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale》提出了Vision Transformer，模型在大规模数据集上进行训练，最终取得了很好的成绩。
缺点：high computational cost of the multiheaded self-attention (MHA).

结合CNN和transformer在边缘设备上运用的探索：MobileViT：consider transformers as convolution and propose a MobileViT block for local-global image context fusion.
缺点：MHA is still the main efficiency bottleneck in this model, especially for the number of MAdds and the inference time on edge devices.

作者的探索：

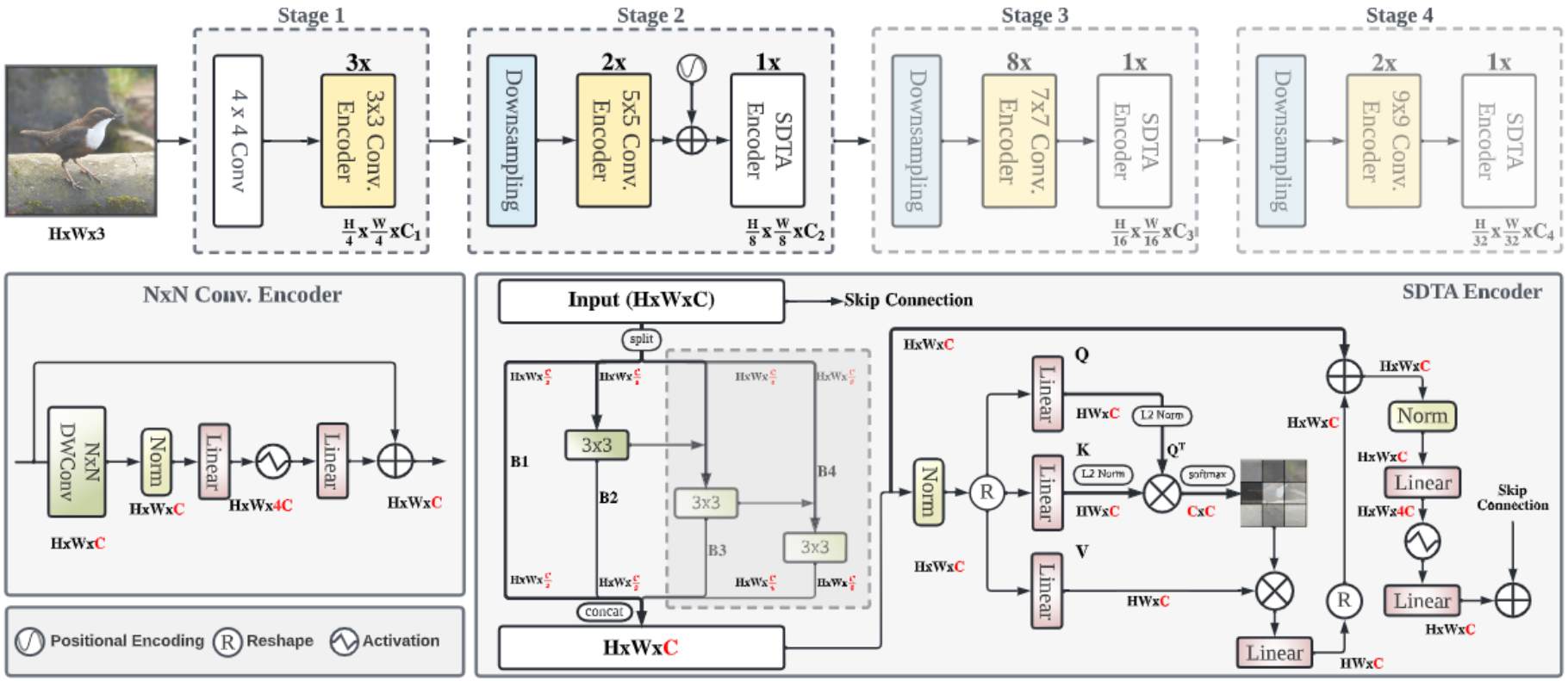
目标：develop a lightweight hybrid design that effectively fuses the merits of ViTs and CNNs for low-powered edge devices.

灵感来源：两个理想中的特性：

1. Encoding the global information efficiently
   做法：Using cross-covariance attention to incorporate the attention operation across the feature channel dimension instead of the spatial dimension within a relatively small number of network blocks.
   好处：Reducing the complexity of the original self-attention operation from quadratic to linear

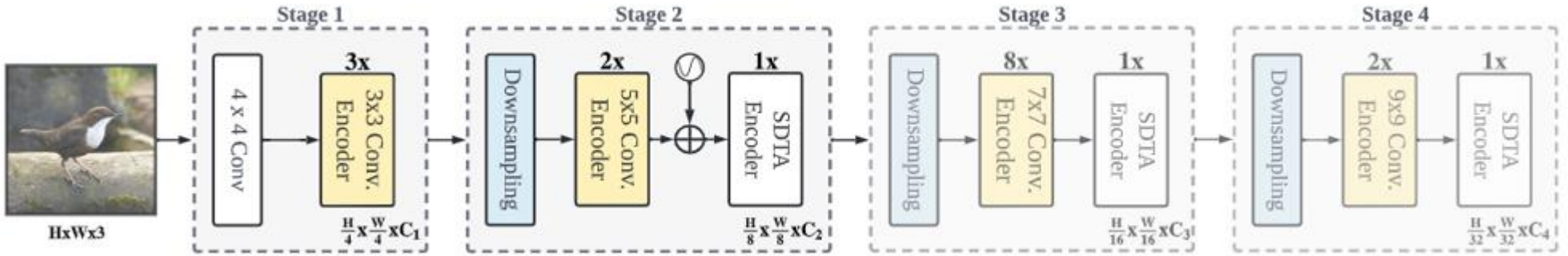
2. Adaptive kernel sizes
   Although large-kernel convolutions have large receptive fields, its parameters and FLOPS quadratically increases as kernel size grows.
   做法：Using smaller kernels at the early stages, while larger kernels at the latter stages.
   优点：capturing low-level features in early stage, and capturing high-level features in later stage.

网络整体结构:



其中，核心为adaptive N × N Conv Encoder和SDTA(split depth-wise transpose attention) Encoder.

模型分析:



论文采用了标准的“四阶段”金字塔式设计，在四个阶段分别以四个不同的尺度提取层次特征。

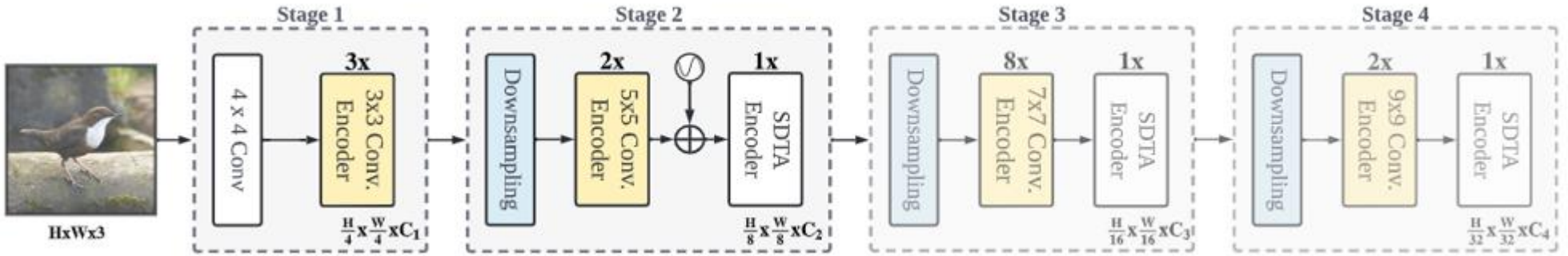
Stage1:

输入大小为H×W×3的特征图像，在网络起始部分，采用了与ViT类似的Patchify策略，被首先传入pathify stem layer（使用non-overlapping卷积，再接一个layer norm）进行下采样，得到$\frac{H}{4} \times \frac{W}{4} \times 3$的特征图；紧接着使用连续堆叠的3个核大小为3× 3的Conv Encoder来提取局部特征。

Stage2:

将上一阶段的输出首先输入至一个Downsampling层，采用步长为2× 2的卷积层实现，得到$\frac{H}{8} \times \frac{w}{8} \times C2$的特征图，再经过2个核大小为5× 5的Conv Encoder。在进入SDTA块之前，在以elemet-wise的方式添加position encoding。Position encoding对于模型性能的影响非常敏感，所以整个模型中只有此处加入了对位置的编码。然后再输入至SDTA Encoder中。

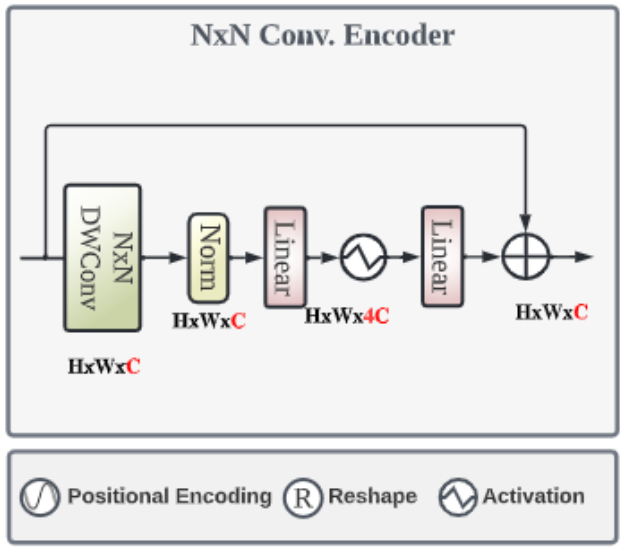
模型分析:



Stage3和4:

同stage2，经downsampling后，输入至Conv Encoder和SDTA Encoder中，分别得到$\frac{H}{16} \times \frac{W}{16} \times C3$和$\frac{H}{32} \times \frac{W}{32} \times C4$的特征图。不同的是没有经过position encoding，每次选用的核的大小不同。

$N \times N$ Conv Encoder：



借鉴了depth-wise separable convolution. 主要由两部分组成：

1. depth-wise convolution with adaptive kernel size
   当nn.Conv2d中的groups=in_channels，out_channels=k×groups，其中k为正整数时，成为depth-wise convolution。

2. point-wise convolution layer
   实则为1×1卷积，丰富局部特征表示，同时也可用来改变输出的通道数。

这种算法可以大幅减少参数量。

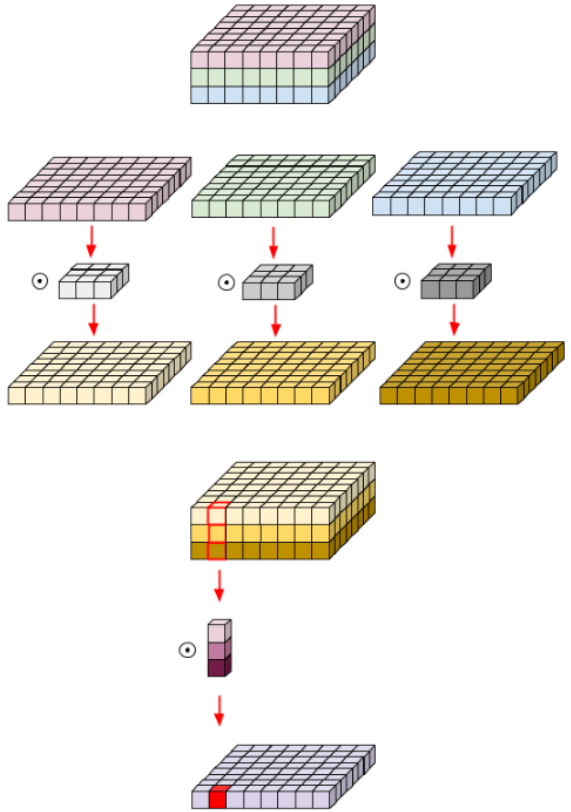
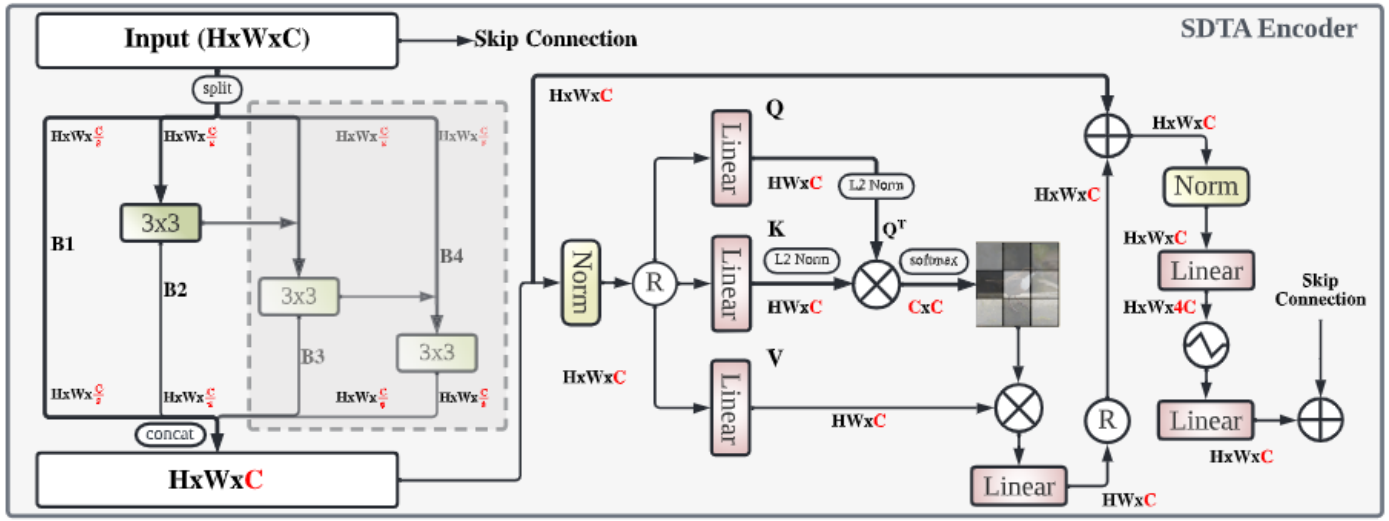


Fig 4. Depth-wise separable convolution

具体到该encoder，经过depth-wise convolution后，经过layer norm层，然后才经过point-wise convolution layer，又经过一个Gaussian Error Linear Unit(GELU)激活函数用来feature mapping，然后再次经过一个point-wise convolution layer，最后使用一个skip connection来使得信息在网络层次中流动。公式如下：

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + Linear_G(Linear(LN(Dw(\boldsymbol{x}_i))))$$

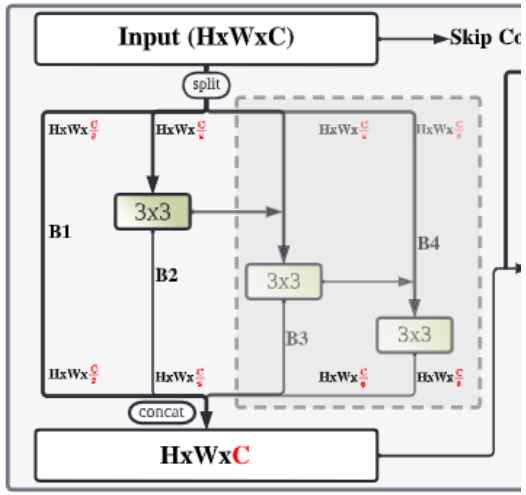
SDTA(split depth-wise transpose attention) Encoder：



主要由两个模块组成：特征编码模块，自注意力计算模块。

The first component strives to learn an adaptive multi-scale feature representation by encoding various spatial levels within the input image and the second part implicitly encodes global image representations.

SDTA(split depth-wise transpose attention) Encoder：

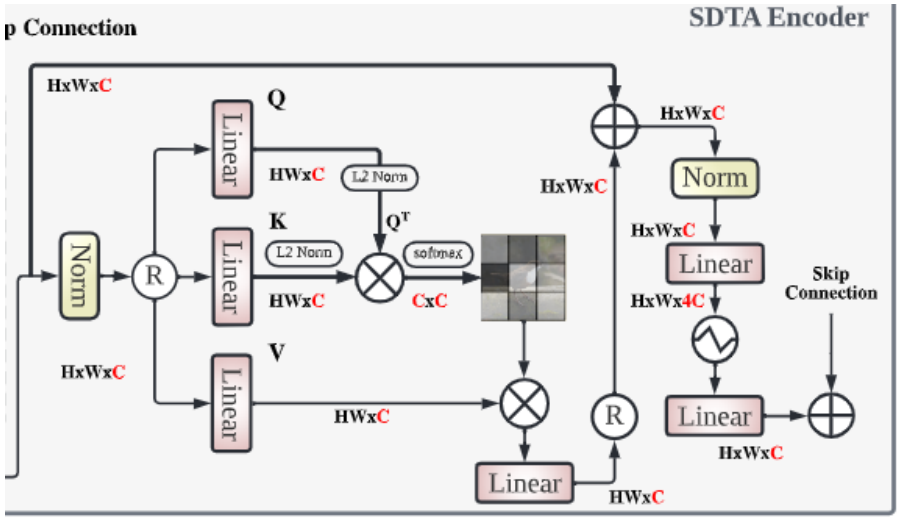


特征编码模块借鉴了Res2Net的思想，希望能获得具有更灵活和自适应空间感受野的输出特征。

首先输入特征通过直接的通道切分被划分为s个子集（图中s=4），每个子集的尺寸均为$H \times W \times \frac{C}{s}$，接着每个子集的计算方式都是将上一个子集的输出特征融合后再经过3×3的depth-wise卷积（除了第一个子集）。最终s个子集的输出特征被拼接后得到具有多尺度感受野的输出特征。

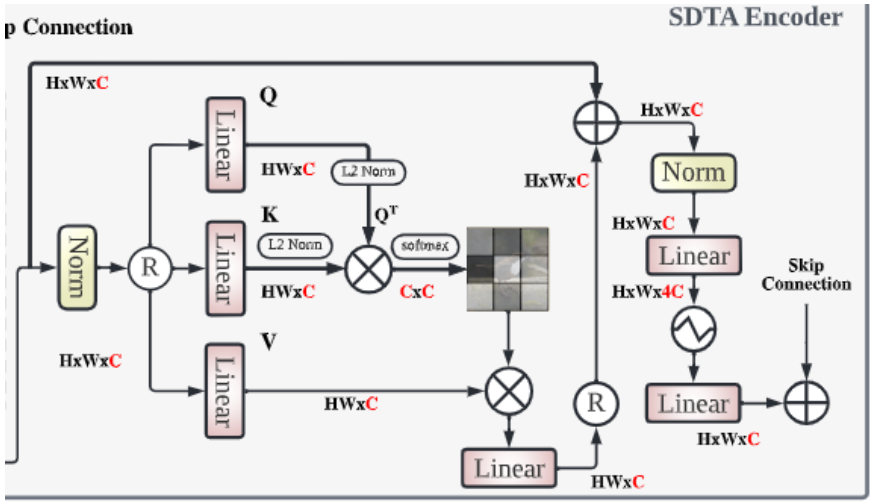
$$y_i = \begin{cases} x_i & i = 1; \\ d_i(x_i) & i = 2, t = 2; \\ d_i(x_i + y_{i-1}) & 2 < i \leq s, t. \end{cases}$$

SDTA(split depth-wise transpose attention) Encoder：



The models use **transposed** query and key attention feature maps in the SDTA encoder. This operation has a **linear complexity** by applying the dot-product operation of the MSA across channel dimensions instead of the spatial dimension, which allows us to compute cross-covariance across channels to generate attention feature maps that have implicit knowledge about the global representations.

SDTA(split depth-wise transpose attention) Encoder：



具体而言，给定一个$H \times W \times C$的归一化的tensor Y，使用三个线性层计算Q，K，V投影，即$\mathrm{Q} = W^{Q}Y, K = W^{K}Y, V = W^{V}Y$，维度是$\mathrm{HW} \times C$；然后在计算cross-covariance attention之前，将L2范数应用于K和V，因为它可以稳定训练过程。然后我们在Q^{T}和K 之间的空间维度上应用点积，$(C \times HW) \cdot (\mathrm{HW} \times C)$，得到$(C \times C)$ softmax scaled attention score matrix，然后再乘V累加。
公式如下：

$$\hat{\boldsymbol{X}} = Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) + \boldsymbol{X}, \qquad (3)$$

$$s.t., \ Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \boldsymbol{V} \cdot \mathtt{softmax}(\boldsymbol{Q}^{T} \cdot \boldsymbol{K})$$

SDTA(split depth-wise transpose attention) Encoder：

本文中attention的计算：

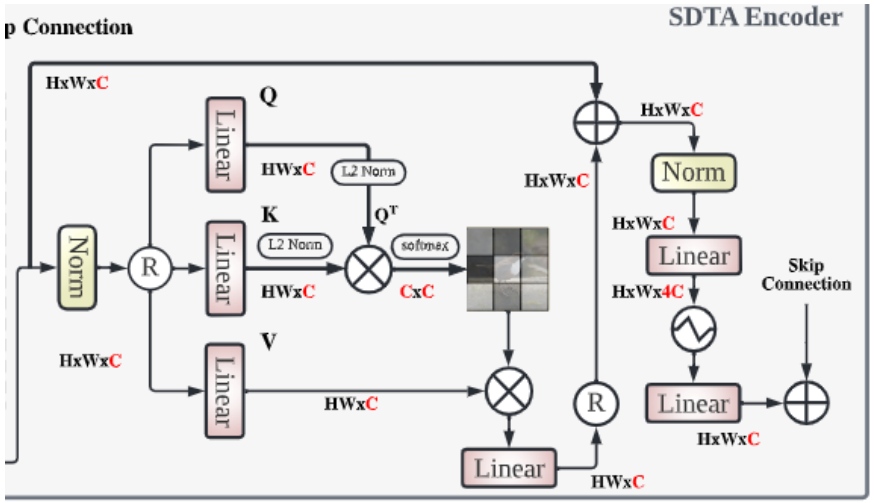
$$\hat{\boldsymbol{X}} = Attention(\boldsymbol{Q},\ \boldsymbol{K},\ \boldsymbol{V}) + \boldsymbol{X}, \tag{3}$$

$$s.t.,\ Attention(\boldsymbol{Q},\ \boldsymbol{K},\ \boldsymbol{V}) = \boldsymbol{V} \cdot \mathtt{softmax}(\boldsymbol{Q}^T \cdot \boldsymbol{K})$$

在《Attention is All You Need》中attention的计算：

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

SDTA(split depth-wise transpose attention) Encoder：



之后，使用两个 1×1 逐点卷积层、LN 和 GELU 激活来生成非线性特征。

EdgeNeXt结构：

Layer	Output Size	#Layers (n)	Kernel	Output Channels		
				XXS	**XS**	**S**
Image	256×256	1	-	-	-	-
Stem	64×64	1	4×4	24	32	48
Conv. Encoder	64×64	3	3×3	24	32	48
Downsampling	32×32	1	2×2	48	64	96
Conv. Encoder	32×32	2	5×5	48	64	96
STDA Encoder	32×32	1	-	48	64	96
Downsampling	16×16	1	2×2	88	100	160
Conv. Encoder	16×16	8	7×7	88	100	160
STDA Encoder	16×16	1	-	88	100	160
Downsampling	8×8	1	2×2	168	192	304
Conv. Encoder	8×8	2	9×9	168	192	304
STDA Encoder	8×8	1	-	168	192	304
Global Average Pooling	1×1	1	-	-	-	-
Linear	1×1	1	-	1000	1000	1000
Model MAdds				0.3G	0.5G	1.3G
Model Prameters				1.3M	2.3M	5.6M

总结：

transformer是一个很成功的模型，其在NLP和CV领域都取得了巨大的成功。但transformer模型过于复杂，在一些资源有限的设备上推理速度太慢。其中，大部分的计算负载都来自于self-attention的计算。

作者所提出的混合模型结合了CNN和transformer的优点，又舍弃了它们的缺点，不仅能对局部和全局信息进行建模，还有较少的参数量和FLOPS，并且结果在多个数据集上表现良好。

总结：

不理解的地方：

1. 在SDTA encoder部分，作者提出：we use transposed query and key attention feature maps in our SDTA encoder. This operation has a **linear complexity** by applying the dot-product operation of the MSA across channel dimensions instead of the spatial dimension…… 一直不是很理解为什么这里是线性的。

   后续补充：《XCiT: Cross-Covariance Image Transformers》这篇论文有解释，但我暂时还没细看。

总结：

学到的知识：

1. 对transformer结构有了简单的认识，以及它在CV领域中的应用情况；
2. 对CNN的优缺点有了更深刻的认识；
3. 如何根据灵感一步步设计自己的模型，通过实验一步步验证自己的模型；

谢谢