视觉 Transformer 研究的关键问题: 现状及展望

田永林 [1,2] 王雨桐 [2] 王建功 [2] 王 晓 [2,3] 王飞跃 [2,3]

摘 要 Transformer 所具备的长距离建模能力和并行计算能力使其在自然语言处理领域取得了巨大成功并逐步拓展至计算机视觉等领域. 本文以分类任务为切入, 介绍了典型视觉 Transformer 的基本原理和结构, 并分析了 Transformer 与卷积神经网络在连接范围、权重动态性和位置表示能力三方面的区别与联系; 同时围绕计算代价、性能提升、训练优化以及结构设计四个方面总结了视觉 Transformer 研究中的关键问题以及研究进展; 并提出了视觉 Transformer 的一般性框架; 然后针对检测和分割两个领域, 介绍了视觉 Transformer 在特征学习、结果产生和真值分配等方面给上层视觉模型设计带来的启发和改变; 并对视觉 Transformer 未来发展方向进行了展望.

关键词 视觉 Transformer, 图像分类, 目标检测, 图像分割, 计算机视觉

引用格式

DOI 10.16383/j.aas.c220027

Key Problems and Progress of Vision Transformers: The State of the Art and Prospects

TIAN Yong-Lin[1,2] WANG Yu-Tong[2] WANG Jian-Gong[2] WANG Xiao[2,3] WANG Fei-Yue[2,3]

Abstract Due to its long-range sequence modeling and parallel computing capability, Transformers have achieved significant success in natural language processing and are gradually expanding to computer vision area. Starting from image classification, we introduce the architecture of classic vision Transformers and compare it with convolutional neural networks in connection range, dynamic weights and position representation ability. Then, we summarize existing problems and corresponding solutions in vision Transformers including computational efficiency, performance improvement, optimization and architecture design. Besides, we propose a general architecture of Vision Transformers. For object detection and image segmentation, we discuss Transformer-based models and their roles on feature extraction, result generation and ground-truth assignment. Finally, we point out the development trends of vision Transformers.

Key words Vision Transformers, image classification, object detection, image segmentation, computer vision

Citation

深度神经网络 (Deep neural network, DNN) 由于其突出的性能表现, 已经成为人工智能系统的主流模型之一[1,2]. 针对不同的任务, DNN 发展出了不同的网络结构和特征学习范式. 其中, 卷积神经网络 (Convolutional neural network, CNN)[3–5] 通过卷积层和池化层等具备平移不变性的算子处理图像数据; 循环神经网络 (Recurrent neural network, RNN)[6,7] 通过循环单元处理序列或时序数据. Transformer[8] 作为一种新的神经网络结构, 目前已被证实可以应用于自然语言处理 (Natural language processing, NLP)、计算机视觉 (Computer vision, CV) 和多模态等多个领域, 并在各项任务中展现出了极大的潜力.

Transformer[8] 兴起于 NLP 领域, 它的提出解决了循环网络模型, 如长短期记忆 (Long short-term memory, LSTM)[6] 和门控循环单元 (Gate recurrent unit, GRU)[7] 等存在的无法并行训练, 同时需要大量的存储资源记忆整个序列信息的问题. Transformer[8] 使用一种非循环的网络结构, 通过编码器—解码器以及自注意力机制[9–12] 进行并行计算, 大幅缩短了训练时间, 实现了当时最优的机器翻译性能. Transformer 模型与循环神经网络以及递归神经网络均具备对序列数据的特征表示能力, 但

收稿日期 XXXX-XX-XX 录用日期 XXXX-XX-XX
Manuscript received Month Date, Year; accepted Month Date, Year
广东省重点领域研发计划 (2020B090921003), 广州市智能网联汽车重大科技专项 (202007050002), 国家自然科学基金项目联合基金 (U1811463) 和英特尔智能网联汽车大学合作研究中心 ("ICRI-IACV") 资助
Supported by Key-Area Research and Development Program of Guangdong Province (2020B090921003), Key Research and Development Program of Guangzhou (202007050002), National Natural Science Foundation of China (U1811463) and Intel Collaborative Research Institute for Intelligent and Automated Connected Vehicles ("ICRI-IACV")
本文责任编委 XXX
Recommended by Associate Editor BIAN Wei
1. 中国科学技术大学自动化系 合肥 230027 2. 中国科学院自动化研究所复杂系统管理与控制国家重点实验室 北京 100190 3. 青岛智能产业技术研究院 青岛 266000
1. Department of Automation, University of Science and Technology of China, Hefei 230027, China 2. The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China 3. Qingdao Academy of Intelligent Industries, Qingdao 266000, China

Transformer 打破了序列顺序输入的限制, 以一种并行的方式建立不同词符间的联系. 基于 Transformer 模型, BERT[13] 在无标注的文本上进行了预训练, 最终通过精调输出层, 在 11 项 NLP 任务中取得了最优表现. 受 BERT 启发, 文献[14]预训练了一个名为 GPT-3 的拥有 1 750 亿个参数的超大规模 Transformer 模型, 在不需要进行精调的情况下, 这一模型在多种下游任务中表现出强大的能力. 这些基于 Transformer 模型的工作, 极大地推动了 NLP 领域的发展.

Transformer 在 NLP 领域的成功应用, 使得相关学者开始探讨和尝试其在计算机视觉领域的应用[15,16]. 一直以来, 卷积神经网络都被认为是计算机视觉的基础模型. 而 Transformer 的出现, 为视觉特征学习提供了一种新的可能[17–21]. 基于 Transformer 的视觉模型在图像分类[15,22,23]、目标检测[16,24]、图像分割[25,26]、视频理解[27,28]、图像生成[29] 以及点云分析[30,31] 等领域取得媲美甚至领先卷积神经网络的效果.

将 Transformer 应用于视觉任务并非一个自然的过程, 一方面, Transformer 网络以序列作为输入形式, 其本身并不直接适用于二维的图像数据[15,16], 将其适配到视觉任务需要经过特殊设计; 另一方面基于全局信息交互的 Transformer 网络往往具有较大的计算量, 同时对数据量也有较高要求, 因此需要考虑其效率以及训练和优化等问题[32,33]. 此外, Transformer 所定义的基于注意力的全局交互机制是否是一种完备的信息提取方式, 来自 CNN 中的经验和技巧能否帮助 Transformer 在计算机视觉任务中取得更好的性能也是需要思考的问题[34,35].

同其它 Transformer 相关的综述文献[17–19]相比, 本文的区别和主要贡献在于我们以视觉 Transformer 在应用过程中存在的关键问题为角度进行切入, 针对每个关键问题组织并综述了相关文章的解决方案和思路, 而其它文献[17–19]则更多是从技术和方法分类的角度入手. 本文梳理了 Transformer 在计算机视觉中应用中的若干关键问题, 同时总结了 Transformer 在计算机视觉的分类、检测和分割任务中的应用和改进. 本文剩余部分组织如下. 第 1 节以 ViT[15] 为例介绍视觉 Transformer 的原理和基本组成, 并对比了 Transformer 与 CNN 的区别和联系, 同时总结了 Transformer 的优势和劣势; 第 2 节介绍 Transformer 研究中的关键问题以及对应的研究进展; 第 3 节给出了视觉 Transformer 的一般性框架; 第 4 节介绍 Transformer 在目标检测领域的应用; 第 5 节介绍 Transformer 在图像分割领域的应用; 第 6 节总结了全文并展望了视觉 Transformer 的发展趋势.

1 ViT 原理介绍与分析

ViT[15] 将 Transformer 结构完全替代卷积结构完成分类任务, 并在超大规模数据集上取得了超越 CNN 的效果[36–39]. ViT 结构如图 1 所示, 它首先将输入图像裁剪为固定尺寸的图像块, 并对其进行线性映射后加入位置编码, 输入到标准的 Transformer 编码器. 为了实现分类任务, 在图像块的嵌入序列中增加一个额外的可学习的类别词符 (Class token) .

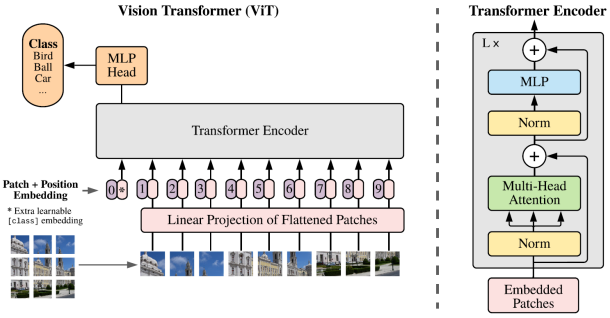


图 1 ViT 模型结构[15]

Fig. 1 The framework of ViT[15]

1.1 图像序列化

对于 NLP 任务, Transformer 的输入是一维的词符嵌入向量, 而视觉任务中, 需要处理的是二维的图像数据. 因此, ViT[15] 首先将尺寸为 $H \times W \times C$ 的图像 $x \in \mathbb{R}^{H\times W\times C}$ 裁剪为 $N = HW/P^2$ 个尺寸为 $P \times P \times C$ 的图像块, 并将每个图像块展开成一维向量, 最终得到 $x_p \in \mathbb{R}^{N\times(P^2\times C)}$. 记 d 为 Transformer 输入嵌入向量的维度, ViT[15] 将 x_p 进行线性映射, 并与类别词符一起组成为 d 维 z_0, 如式 (1) 所示, 作为 Transformer 编码器的输入.

$$z_0 = [x_{class}; x_p^1 E; x_p^2 E; \cdots; x_p^N E] + E_{pos}, \quad E \in \mathbb{R}^{(P^2\times C)\times d}, E_{pos} \in \mathbb{R}^{(N+1)\times d} \tag{1}$$

其中, $z_0^0 = x_{class}$ 是为了实现分类任务加入的可学习的类别词符, E 是实现线性映射的矩阵, E_{pos} 是位置编码. 类别词符以网络参数的形式定义, 其本身是一种网络权重, 可以通过梯度进行更新。类别词符 z_0^0 本身不具备当前输入的特征和信息, 而是在与图像块词符串联后通过自注意力机制实现对图像特征的信息交互或信息聚合, 在编码器最后一层之后, 类别词符 z_L^0 作为对图像特征的聚合, 被送入分类头进行类别预测.

1.2 编码器

ViT[15] 的编码器由 L (ViT[15] 中, $L \in \{12, 24, 32\}$) 个相同的层堆叠而成, 每个层又由两个子层组成. 其中, 第一个子层是多头自注意力机

制（Multi-head self-attention, MSA），第二个子层是多层感知机（Multi-layer perceptron, MLP）. 在数据进入每个子层前, 都使用层归一化（Layer normalization, LN）[40] 进行归一化处理, 数据经过每个子层后, 又使用残差连接与输入进行直接融合. 值得注意的是, 为了实现残差连接[5], ViT 编码器的每一层的输出维度都设计为 d 维. 最后, 经过 L 层网络编码之后, 类别词符 z_L^0 被送入到由 MLP 构成的分类头中, 从而预测得到图像的类别 y. 第 l 层的特征计算过程如下:

$$z_l' = \mathrm{MSA}(\mathrm{LN}(z_{l-1})) + z_{l-1} \quad , \quad l = 1, \cdots, L \tag{2}$$

$$z_l = \mathrm{MLP}(\mathrm{LN}(z_l')) + z_l' \quad , \quad l = 1, \cdots, L \tag{3}$$

类别预测结果的产生可表示为:

$$y = \mathrm{LN}(z_L^0) \tag{4}$$

1.3 注意力机制

注意力机制（Attention）最早应用于 NLP 任务中[9,12,41], 通过引入长距离上下文信息, 解决长序列的遗忘现象. 在视觉任务中, 注意力机制同样被用来建立空间上的长距离依赖, 以解决卷积核感受野有限的问题[42,43].

ViT 使用的自注意力机制（Self-attention, SA）是一种缩放点积注意力（Scaled dot-product attention）, 其计算过程如图 2 所示. 自注意力层通过查询（Query）与键（Key）- 值（Value）对之间的交互实现信息的动态聚合. 对输入序列 $z \in \mathbb{R}^{N\times d}$, 通过线性映射矩阵 U_{QKV} 将其投影得到 Q、K 和 V 三个向量. 在此基础上, 计算 Q 和 K 间的相似度 A, 并根据 A 实现对 V 进行加权. 自注意力的计算过程如下所示:

$$[Q, K, V] = zU_{QKV}, U_{QKV} \in \mathbb{R}^{d\times 3d_h} \tag{5}$$

$$A = \mathrm{softmax}(QK^T/\sqrt{d_h}), A \in \mathbb{R}^{N\times N} \tag{6}$$

加权聚合过程可表示为:

$$SA(z) = AV \tag{7}$$

为了提高特征多样性, ViT 使用了多头自注意力机制. 多头自注意力层使用多个自注意力头来并行计算, 最后通过将所有注意力头的输出进行拼接得到最终结果. 多头注意力计算过程如下所示:

$$\mathrm{MSA}(z) = [\mathrm{SA}_1(z); \mathrm{SA}_2(z); \cdots; \mathrm{SA}_h(z)]U_{msa} \tag{8}$$

其中, $U_{msa} \in \mathbb{R}^{h\cdot d_h\times d}$ 为映射矩阵, 用于对拼接后的特征进行聚合, h 表示自注意力头的个数, d_h 为每个自注意力头的输出维度. 为了保证在改变 h 时模型参数量不变, 一般将 d_h 设置为 d/h. 多头自注意力机制中并行使用多个自注意力模块, 可以丰富注意力的多样性, 从而增加模型的表达能力.

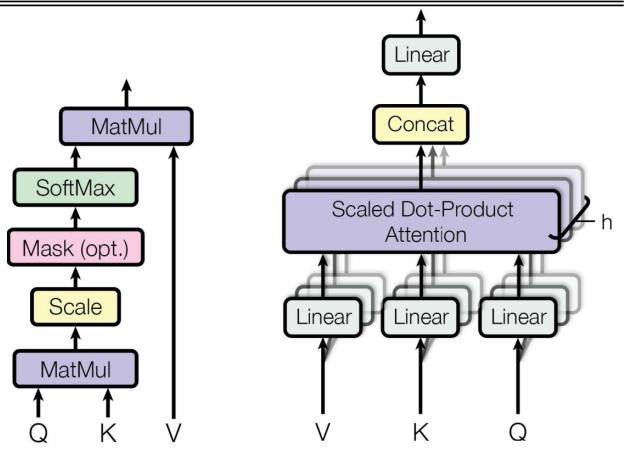


图 2 自注意力[15] 与多头自注意力[15]

Fig. 2 Self-attention[15] and multi-head self-attention[15]

1.4 位置编码

ViT 使用了可学习的位置编码方式, 通过定义可训练变量实现位置编码. 相比之下, 一种更为原始的位置编码方案[8] 是使用正余弦函数实现:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d}) \tag{9}$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d}) \tag{10}$$

其中, pos 是每个图像块在图像中的位置, $i \in [0, \cdots, d/2]$ 用于计算通道维度的索引. 对于同一个 i, 通道上第 $2i$ 和 $2i+1$ 个位置的编码是具有相同角速度的正弦和余弦值. 为了使得位置编码可以与输入嵌入相加, 位置编码需要与嵌入保持相同的维度.

1.5 Transformer 同卷积神经网络的区别与联系

本节主要从连接范围[44]、权重动态性[44] 和位置表示能力三个方面来阐述 Transformer 同卷积神经网络的区别与联系.

1.5.1 连接范围

卷积神经网络构建在输入的局部连接之上, 通过不断迭代, 逐渐扩大感受野, 而 Transformer 则具备全局交互机制, 其有效感受野能够迅速扩大. 图 3 展示了语义分割任务中, DeepLabv3+[45] 和 SegFormer[25] 在有效感受野上的对比, 可以看到, 相比于卷积神经网络, Transformer 网络的有效感受野范围具备明显优势. 虽然卷积核的尺寸可以设置为全图大小, 但这种设置在图像数据处理中并不常见, 因为这将导致参数量的显著增加.

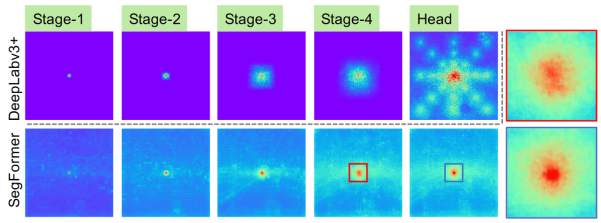


图 3 Transformer 与 CNN 有效感受野对比[25]

Fig. 3 The comparison[25] of effective receptive field between Transformer and CNN

1.5.2 权重动态性

传统卷积神经网络在训练完成后, 卷积核权重不随输入或滑动窗口位置变化而改变[46], 而 Transformer 网络通过相似性度量动态地生成不同节点的权重并进行信息聚合. Transformer 的动态性与动态卷积[46] 具备相似的效果, 都能响应输入信息的变化.

1.5.3 位置表示能力

Transformer 使用序列作为输入形式, 其所使用的自注意力机制和通道级 MLP 模块均不具备对输入位置的感知能力, 因此 Transformer 依赖位置编码来实现对位置信息的补充. 相比之下, 卷积神经网络处理二维图像数据, 一方面卷积核中权重的排列方式使其具备了局部相对位置的感知能力, 另一方面, 有研究表明[47], 卷积神经网络使用的零填充 (Zero padding) 使其具备了绝对位置感知能力, 因此, 卷积神经网络不需要额外的位置编码模块.

1.6 ViT 的优劣势分析

ViT[15] 模型的优势在于其构建了全局信息交互机制, 有助于建立更为充分的特征表示. 此外, ViT 采用了 Transformer 中标准的数据流形式, 有助于同其它模态数据进行高效融合. ViT 存在的问题主要在三个方面, 首先全局注意力机制计算量较大, 尤其是面对一些长序列输入时, 其与输入长度成平方的计算代价极大地限制了其在高分辨率输入和密集预测任务中的应用; 其次, 不同于卷积中的局部归纳偏置, ViT 模型从全局关系中挖掘相关性, 对数据的依赖较大, 需要经过大量数据的训练才能具备较好效果; 此外, ViT 模型的训练过程不稳定且对参数敏感.

2 Transformer 研究中的关键问题

本节以图像分类这一基本的视觉任务为切入, 着重介绍 Transformer 在用于视觉模型骨架时的关键研究问题以及对应的研究进展.

2.1 如何降低 Transformer 的计算代价

Transformer 的设计使其具有全局交互能力, 但同时其全局自注意力机制也带来了较高的时间和空间代价, 如何设计更高效的 Transformer 机制成为研究热点之一[48]. 原始的 Transformer 使用了点积注意力机制 (Dot-product attention), 其具有二次时间和空间复杂度, 因此不利于推广到高分辨率图像和特征的处理中. 现有文献主要从输入和注意力设计两个角度来降低 Transformer 注意力机制的复杂度. 表 1 总结了多种 Transformer 模型的自注意力机制的计算复杂度.

2.1.1 受限输入模式

减少输入到注意力层的序列的长度是降低计算量的直接手段, 现有文献主要从输入下采样、输入局部化和输入稀疏化三个角度来限制序列的长度[49].

1) 输入下采样: PVT[22] 通过金字塔型的网络设计将图像分辨率层级尺度衰减, 来逐渐降低图像序列的长度. DynamicViT[51] 通过输入学习动态的序列稀疏化策略, 以此逐渐降低图像序列长度. 该类方法在维持全局交互的基础上, 以减小分辨率的形式实现对计算量的降低.

2) 输入局部化: 输入局部化旨在限制注意力的作用范围, 通过设计局部的注意力机制降低计算量, 例如 Swin Transformer[23] 提出了基于窗口的多头注意力机制, 将图像划分成多个窗口, 仅在窗口内部进行交互.

3) 输入稀疏化: 稀疏化通过采样或压缩输入来降低注意力矩阵的尺寸, 例如, CrossFormer[52] 提出了对输入进行间隔采样来构建长距离注意力 (Long distance attention). Deformable DETR[24] 将可形变卷积的设计引入到注意力的计算中, 通过学习采样点的位置信息实现稀疏交互机制, 在减小计算量的同时维持了较大范围的感受野.

2.1.2 高效注意力机制

核函数方法[33] 和低秩分解[53] 是用来降低注意力复杂度的主要方法[48]. 表 1 中总结了不同注意力机制的时间复杂度和空间复杂度, 同时我们给出了卷积算子的复杂度作为参考. 为了方便对比, 我们在卷积复杂度的计算中, 将特征图的长宽乘积等同于 Transformer 的输入序列长度, 将 Transformer 的词符特征的维度视为卷积输入与输出通道数, 将局部 Transformer 的窗口大小 s 视为卷积核大小.

1) 核函数方法 (Kernelization): 核函数方法通过重构注意力计算机制打破归一化函数对 Q 和 K 计算的绑定, 来降低注意力计算的时间和空间成本[33,54,55]. 点积注意力机制可被表示为如下形式:

$$\boldsymbol{D}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \rho\left(\boldsymbol{Q}\boldsymbol{K}^T\right)\boldsymbol{V} \tag{11}$$

表 1 不同 Transformer 自注意力机制以及卷积的时间和空间复杂度 (N, d, s 分别表示序列长度、特征维度和局部窗口尺寸. 表中数据主要参考文献 [56])

Table 1 The time and space complexity of different Transformer frameworks (N, d, s denote the length, dimension and local window size respectively. Most of the data in the table are from [56]

)

名称	时间复杂度	空间复杂度
Convolution	$O(Nd^2s)$	-
Transformer [8]	$O(N^2d)$	$O(N^2+Nd)$
Sparse Transformers [49]	$O(N\sqrt{N}d)$	-
Reformer [50]	$O(N\log Nd)$	$O(N\log N+Nd)$
Linear Transformer [33]	$O(Nd^2)$	$O(Nd+d^2)$
Performer [54]	$O(Nd^2\log d)$	$O(Nd\log d+d^2\log d)$
AFT-simple [56]	$O(Nd)$	$O(Nd)$
AFT-full [56]	$O(N^2d)$	$O(Nd)$
AFT-local (1D) [56]	$O(Nsd)$	$O(Nd)$
Swin Transformer (2D) [23]	$O(Nsd)$	-

其中, ρ 表示激活函数, 在经典 Transformer [8] 中, 激活函数为 Softmax. Efficient attention [32] 和 Linear Transformer [33] 将注意力机制的计算转换为式 (12) 的形式, 实现对点积注意力的近似.

$$\boldsymbol{E}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\phi(\boldsymbol{Q})\left(\phi(\boldsymbol{K})^T\boldsymbol{V}\right) \tag{12}$$

这种方式避免了对具有 $\mathcal{O}(N^2)$ 时间和空间复杂度的注意力图的计算和存储, 提高了注意力的计算效率. AFT [56] 采用了类似式 (12) 的设计, 但使用逐元素相乘代替矩阵的点积运算, 从而进一步降低了计算量.

2) 低秩方法 (Low-rank methods): 低秩分解假定了注意力矩阵是低秩的, 因此可以将序列长度进行压缩以减少计算量. 考虑到注意力层输出序列长度只与查询的节点个数有关, 因此通过压缩键和值向量的序列长度, 不会影响最终输入的尺寸. PVT [22]、ResT [53] 和 CMT [34] 利用卷积减少了键和值对应的词符个数以降低计算量. SOFT [57] 使用高斯核函数替换 Softmax 点积相似度, 并通过卷积或池化的方式从序列中采样, 实现对原始注意力矩阵的低秩近似.

2.2 如何提升 Transformer 的表达能力

本小节主要围绕如何提高 Transformer 模型的表达能力而展开, 视觉 Transformer 的研究仍处于起步, 一方面可以借鉴 CNN 的改进思路, 通过类似多尺度等的方案实现对性能的提升, 另一方面由于 Transformer 基于全局信息的交互, 使其具有不同于 CNN 的特征提取范式, 从而为引入 CNN 设计范式进而提升性能提供了可能. CNN 的局部性 (Locality) 设计范式可以丰富 Transformer 网络的特征多样性, 同时也有利于改善 Transformer 特征的过度光滑 (Over-smoothing) 的问题 [59]. 此外, 对 Transformer 本身机制, 如注意力和位置编码等的改进也有望提高其表达能力. 表 2 展示了不同 Transformer 模型在 ImageNet [4] 上的性能对比.

2.2.1 多尺度序列交互

多尺度特征在 CNN 中已经获得了较为广泛的应用 [62], 利用多尺度信息能够很好地结合高分辨率特征和高语义特征, 实现对不同尺度目标的有效学习. 在视觉 Transformer 中, CrossViT [63] 使用两种尺度分别对图像进行划分并独立编码, 对编码后的多尺度特征利用交互注意层实现两种尺度序列之间的信息交互. CrossFormer [52] 则借助金字塔型网络, 在不同层得到不同尺度的特征, 之后融合不同层的特征, 以进行跨尺度的信息交互.

2.2.2 图像块特征多样化

DiversePatch [64] 发现了在 Transformer 的深层网络中, 同层图像块的特征之间的相似性明显增大, 并指出这可能引起 Transformer 性能的退化, 使其性能无法随深度增加而继续提升. 基于该发现,DiversePatch [64] 提出了三种方式来提高特征的多样性. 首先, 对网络最后一层的图像块特征之间计算余弦相似度, 并作为惩罚项加入到损失计算中. 其次, 基于对 Transformer 网络首层图像块

表 2 视觉 Transformer 算法在 ImageNet-1k 上的 Top-1 准确率比较. 表中数据主要参考文献 [18]

Table 2 The comparison of Top-1 accuracy of different visual Transformers on ImageNet-1k dataset. Most of the data in the table are from [18]

方法名称	迭代轮次	批处理大小	参数量 (M)	计算量 (GFLOPs)	图像尺寸 训练	图像尺寸 测试	Top-1 Acc.
ViT-B/16 [15]	300	4 096	86	55.4	224	384	77.9
ViT-L/16 [15]			307	190.7	224	384	76.5
DeiT-Ti [58]	300	1 024	5	1.3	224	224	72.2
DeiT-S [58]			22	4.6	224	224	79.8
DeiT-B [58]			86	17.6	224	224	81.8
DeiT-B↑ [58]			86	52.8	224	384	83.1
ConViT-Ti [60]	300	512	6	1	224	224	73.1
ConViT-S [60]			27	5.4	224	224	81.3
ConViT-B [60]			86	17	224	224	82.4
LocalViT-T [61]	300	1 024	5.9	1.3	224	224	74.8
LocalViT-S [61]			22.4	4.6	224	224	80.8
CeiT-T [73]	300	1 024	6.4	1.2	224	224	76.4
CeiT-S [73]			24.2	4.5	224	224	82.0
CeiT-S↑ [73]			24.2	12.9	224	384	83.3
ResT-Small [53]	300	2 048	13.66	1.9	224	224	79.6
ResT-Base [53]			30.28	4.3	224	224	81.6
ResT-Large [53]			51.63	7.9	224	224	83.6
Swin-T [23]	300	1 024	29	4.5	224	224	81.3
Swin-S [23]			50	8.7	224	224	83.0
Swin-B [23]			88	15.4	224	224	83.3
Swin-B↑ [23]			88	47.0	224	384	84.2
VOLO-D1 [68]	300	1 024	27	6.8	224	224	84.2
VOLO-D2 [68]			59	14.1	224	224	85.2
VOLO-D3 [68]			86	20.6	224	224	85.4
VOLO-D4 [68]			193	43.8	224	224	85.7
VOLO-D5 [68]			296	69.0	224	224	86.1
VOLO-D5↑ [68]			296	304	224	448	87.0
PVT-Tiny [22]	300	128	13.2	1.9	224	224	75.1
PVT-Small [22]			24.5	3.8	224	224	79.8
PVT-Medium [22]			44.2	6.7	224	224	81.2
PVT-Large [22]			61.4	9.8	224	224	81.7
DeepViT-S [66]	300	256	27	6.2	224	224	82.3
DeepViT-L [66]			55	12.5	224	224	83.1
Refined-ViT-S [59]	300	256	25	7.2	224	224	83.6
Refined-ViT-M [59]			55	13.5	224	224	84.6
Refined-ViT-L [59]			81	19.1	224	224	84.9
Refined-ViT-L↑ [59]		512	81	69.1	224	384	85.7
CrossViT-9 [63]	300	4 096	8.6	1.8	224	224	73.9
CrossViT-15 [63]			27.4	5.8	224	224	81.5
CrossViT-18 [63]			43.3	9.0	224	224	82.5

特征多样性较高的观察, DiversePatch 提出使用对比损失（Contrastive loss）来最小化同一图像块在首层和尾层对应特征的相似性, 而最大化不同图像块在首层和尾层对应特征的相似性. 最后, 基于 CutMix[65] 的思想, DiversePatch 提出了混合损失（Mixing loss）, 通过将来自不同图片的图像块进行混合, 使网络学习每个图像块的类别, 以避免特征同质化.

2.2.3 注意力内容多样化

DeepViT[66] 观察到 Transformer 中的注意力坍塌（Attention collapse）现象, 即随着网络加深, 深层注意力图不同层之间的相似性逐渐增大甚至趋同, 并指出注意力相似性增加和特征图相似性增加有密切关系, 从而导致了 Transformer 性能随层数增加而快速饱和. 为了避免注意力坍塌现象, DeepViT 提出了增加词符的嵌入维度的方法和重注意力（Re-attention）机制. 增加词符的嵌入维度有助于词符编码更多信息, 从而提高注意力的多样性, 但同时会带来参数量的显著增加. 重注意力机制基于层内多头注意力的多样性较大的现象, 通过对多头注意力以可学习的方式进行动态组合来提高不同层注意力的差异. 重注意力机制 $\boldsymbol{R}$ 可表示为式 (13) 的形式, 其中 $\Theta \in \mathbb{R}^{h\times h}$ 表示一个可学习的变换矩阵，用于对多头注意力进行重组.

$$\boldsymbol{R}(Q,K,V) = \mathrm{Norm}\left(\Theta^T\left(\rho\left(QK^T\right)\right)\right)V \tag{13}$$

Refiner[59] 基于类似的思想提出了注意力扩张（Attention expansion）和注意力缩减（Attention reduction）模块, 通过学习多头注意力的组合方式来构建多样化的注意力, 并可灵活拓展注意力的个数. 同时, Refiner 提出使用卷积来增强注意力图的局部特征, 从而降低注意力图的光滑程度.

2.2.4 注意力形式多样化

经典 Transformer 中的注意力机制依赖点对间的交互来计算其注意力, 其基本作用是实现自我对齐, 即确定自身相对于其它节点信息的重要程度[67]. Synthesizer[67] 指出这种通过点对交互得到的注意力有用但却并不充分, 通过非点对注意力能够实现对该交互方式的有效补充.

1) 非点对注意力（Non-token-token attention）: Synthesizer[67] 提出了两种新的非点对注意力实现方法, 即基于独立词符和全局任务信息的注意力计算方法. 基于独立词符的注意力, 以每一个词符为输入, 在不经过与其它词符交互的情况下, 学习其它词符相对于当前词符的注意力; 基于全局任务信息的注意力生成方法则完全摆脱注意力对当前输入的依赖, 通过定义可训练参数从全局任务信息中学习注意力. 这两种方式可视为从不同的角度来拓展注意力机制, 实验验证了它们同基于点对的注意力能形成互补关系. VOLO[68] 同样提出了基于独立词符的注意力生成方法, 并将注意力的范围限制在局部窗口内, 形成了类似动态卷积的方案.

2.2.5 Transformer 与 CNN 的结合

局部性是 CNN 的一个典型特征, 它是基于临近像素具有较大相关性的假设而形成的一种归纳偏置（Inductive bias）[69–71]. 相比之下, Transformer 的学习过程基于全局信息的交互, 因此在学习方式和特征性质等方面与 CNN 存在一定差异[72], 将 CNN 与 Transformer 进行结合有助于提升 Transformer 网络对特征的学习和表达能力[23,58,73,74]. 本节将从机理融合、结构融合和特征融合三个角度介绍 CNN 与 Transformer 结合的工作.

1) 机理融合: 该方式通过在 Transformer 网络的设计中引入 CNN 的局部性来提高网络表达能力. 以 Swin Transformer[23] 为代表的 Transformer 网络通过将注意力限制在局部窗口内, 来显式地进行局部交互. 此外, CeiT[73] 在 FFN 模块中, 引入局部特征学习, 以建模局部关系.

2) 结构融合: 这种融合方法通过组合 Transformer 和 CNN 的模块形成新的网络结构. CeiT[73] 和 ViTc[35] 将卷积模块添加到 Transformer 前实现对底层局部信息的提取. MobileViT[75] 将 Transformer 视为卷积层嵌入到卷积神经网络中, 实现了局部信息和全局信息的交互.

3) 特征融合: 该方式在特征级别实现对 CNN 特征和 Transformer 特征的融合. 这类方法往往采用并行的分支结构, 并将中间特征进行融合交互. MobileFormer[74] 和 ConFormer[76] 采用并行的 CNN 和 Transformer 分支, 并借助桥接实现特征融合. DeiT[58] 借助知识蒸馏的思路, 通过引入蒸馏词符（Distillation token）来将 CNN 的特征引入到 Transformer 的学习过程中.

2.2.6 相对位置编码

原始 Transformer 使用绝对位置编码为输入词符提供位置信息, 只能隐式地度量相对位置信息[77]. 相对位置编码（Relative position encoding, RPE）则直接对序列的距离进行表示, 能够实现对不同长度的序列的表达不变性, 同时相关关系的显式度量也有利于提升模型性能[78].

为了说明不同编码方式在自注意力层的表现不同, 这里针对式（6）和式（7）对自注意力机制进一步说明. 对包含 n 个元素 $x_i \in \mathbb{R}^{d_x}$ 的输入序列 $x = (x_1, \cdots, x_n)$, 自注意力的输出序列为 $z = (z_1, \cdots, z_n)$, 其中, 每一个输出元素 $z_i \in \mathbb{R}^{d_z}$

是所有输入元素的加权和, 计算过程如下所示:

$$z_i = \sum_{j=1}^{n} \alpha_{ij}(x_j W^V) \tag{14}$$

其中, 每个权重系数 α_{ij} 通过 Softmax 计算得到:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum\limits_{k=1}^{n} \exp(e_{ik})} \tag{15}$$

其中, e_{ij} 通过缩放点积比较两个输入元素计算得到:

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T}{\sqrt{d_z}} \tag{16}$$

其中, W^Q、W^K 和 $W^V \in \mathbb{R}^{d_x \times d_z}$ 是参数矩阵. RPE 在自注意力机制中加入输入元素间的相对位置信息, 以提升模型表达能力. 本部分后续内容对典型的相对位置编码方法进行了介绍[82].

Shaw 等人提出的 RPE[78]: 基于自注意力的相对位置编码, 将输入词符建模为有向的全连接图, 任意两个位置 i 和 j 间的边为可学习的相对编码向量 p_{ij}^V, p_{ij}^K. 将编码向量嵌入自注意力机制, 计算过程如下所示:

$$z_i = \sum_{j=1}^{n} \alpha_{ij}(x_j W^V + p_{ij}^V) \tag{17}$$

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K + p_{ij}^K)^T}{\sqrt{d_z}} \tag{18}$$

其中, $p_{ij}^K, p_{ij}^V \in \mathbb{R}^{d_z}$ 分别为加在键和值上的可学习的权重参数.

Transformer-XL 的 RPE[79]: 相比 Shaw 的方法, 该方法加入了全局内容和全局位置偏置, 使得在特定长度序列下训练的模型能够泛化到更长的序列输入上. 计算过程如下所示:

$$e_{ij} = \frac{(x_i W^Q + u)(x_j W^K)^T + (x_i W^Q + v)(s_{i-j} W^R)^T}{\sqrt{d_z}} \tag{19}$$

其中, $u, v \in \mathbb{R}^{d_z}$ 替换原始绝对位置信息的可学习向量, $s_{i-j} W^R$ 替换绝对位置信息的相对位置信息. $W^R \in \mathbb{R}^{d_z \times d_z}$ 是可学习的矩阵, s 是正弦编码向量.

Huang 等人提出的 RPE[80]: 相比 Shaw 的 RPE 中只建模了键和查询、查询和相对位置编码的交互, 增加了对键和相对位置交互的显式建模, 使其具有更强的表达能力. 计算过程如下所示:

$$e_{ij} = \frac{(x_i W^Q + p_{ij})(x_j W^K + p_{ij})^T - p_{ij} p_{ij}^T}{\sqrt{d_z}} \tag{20}$$

其中, $p_{ij} \in \mathbb{R}^{d_z}$ 是查询和键共享的相对位置编码.

相比 NLP 任务中输入为一维词符序列的语言模型, 视觉任务中输入为二维图像, 因此需要二维的位置信息.

SASA 中的 RPE[81]: 将二维的相对位置信息分为水平和垂直的两个方向, 在每一个方向进行一维位置编码, 并与特征嵌入相加, 相对位置信息的计算过程如下所示:

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K + concat(p_{\delta x}^K, p_{\delta y}^K))^T}{\sqrt{d_z}} \tag{21}$$

其中, $\delta x = x_i - x_j$ 和 $\delta y = y_i - y_j$ 分别为 x 轴和 y 轴的相对位置偏置, $p_{\delta x}^K$ 和 $p_{\delta y}^K$ 分别为长度为 $\frac{1}{2} d_z$ 的可学习向量, $concat$ 将这两个向量拼接起来组成最终的长度为 d_z 的相对位置编码.

Axial-Deeplab 中的 RPE[129]: 相比 SASA 中的 RPE 只在键上加入偏置, 该方法同时对查询、键和值引入了偏置项. 通过轴向注意力, 将二维的注意力先后沿高度和宽度轴分解为两个一维的注意力.

iRPE (image RPE)[82]: 以往的相对位置编码都依赖于输入嵌入, 为了研究位置编码对输入嵌入的依赖关系, 该方法提出了两种相对位置编码模式, 偏置模式和上下文模式. 偏置模式的相对位置编码不依赖输入嵌入, 上下文模式则考虑了相对位置编码与查询、键和值间的交互. 二者都可以表示为如下形式:

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T + b_{ij}}{\sqrt{d_z}} \tag{22}$$

其中, $b_{ij} \in \mathbb{R}$ 是决定偏置和上下文模式的二维相对位置编码. 偏置模式下表示为如下形式:

$$b_{ij} = r_{ij} \tag{23}$$

其中, r_{ij} 是可学习的标量, 表示位置 i 和 j 间的距离. 上下文模式下表示为如下形式:

$$b_{ij} = (x_i W^Q) r_{ij}^T \tag{24}$$

其中, $r_{ij} \in \mathbb{R}^{d_z}$ 是与键相加的可学习偏置向量. 在 ImageNet[83] 上使用 DeiT-S[58] 完成分类任务发现, 上下文模式比偏置模式具有更好的表达能力.

同时, 为了研究相对位置的方向性是否有助于视觉任务, 设计了不同的相对位置映射函数以实现无方向性位置编码和有方向性位置编码. 无方向的映射包括欧式距离法和量化欧式距离法, 都是通过相对位置坐标 $(x_i - x_j, y_i - y_j)$ 的欧式距离计算得到:

$$r_{ij} = p_{I(i,j)} \tag{25}$$

$$I(i,j) = g((\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2})) \tag{26}$$

$$I(i,j)) = g(quant(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2})) \tag{27}$$

其中, 偏置模式下 $p_{I(i,j)}$ 是可学习的标量, 上下文模式下是向量. $g(\cdot)$ 是将相对位置映射为权重的分段函数. $quant$ 将具有不同相对位置的邻居映射为不同的值.

方向性的映射位置编码包括交叉法和乘积法, 交叉法分别计算横纵方向的位置编码, 并进行相加, 其计算过程如下所示:

$$r_{ij} = p^x_{I^x(i,j)} + p^y_{I^y(i,j)} \tag{28}$$

$$I^x(i,j) = g(x_i - x_j) \tag{29}$$

$$I^y(i,j) = g(y_i - y_j) \tag{30}$$

乘积法将两个方向上的位置偏移构成索引对, 进而产生位置编码如下所示:

$$r_{ij} = p_{I^x(i,j),I^y(i,j)} \tag{31}$$

实验发现, 方向性位置编码比非方向性位置编码具有更好的表达能力.

2.3 Transformer 的训练和优化问题

Transformer 的训练过程需要精心设计学习率以及权重衰减等多项参数, 并且对优化器的选择也较为苛刻, 例如其在 SGD 优化器上效果较差[35]. 文献[35]和 CeiT[73] 在图像编码前使用卷积层级来解决 Transformer 的难优化以及参数敏感的问题, 引入卷积后, 模型对学习率和权重衰减等参数的敏感性得到了显著降低, 收敛速度得到加快, 同时在 SGD 优化器上也可以进行稳定的学习. 关于在早期引入卷积机制使模型性能得到改善的原因, Raghu 等[72] 给出了解释和分析, 他们利用充足的数据训练视觉 Transformer, 发现模型在性能提升的同时, 其在浅层也逐步建立了局部表示. 这表明浅层局部表示对性能提升可能有显著的影响, 同时也为解释在浅层引入具备局部关系建模能力的卷积层从而提升 Transformer 的训练稳定性和收敛速度的现象提供了一个思路.

2.4 结构设计问题

本节将从整体结构和局部结构两个角度对 Transformer 方法以及类 Transformer 方法进行介绍. 其中, 整体结构上, 我们以图像特征尺寸变化情况为依据, 将其分为单尺度的直筒型结构和多尺度的金字塔型结构[84]; 在局部结构上, 我们主要围绕 Transformer 中基本特征提取单元的结构, 分析卷积以及 MLP 方法在其中的替代和补充作用以及由此形成的不同局部结构设计.

2.4.1 单尺度和多尺度结构设计

单尺度和多尺度的结构简图如图 4 所示[84], 其中交互模块表示空间或通道级的信息交互层, 聚合层表示对全局信息进行聚合, 例如全局最大值池化或基于类别词符的查询机制等. 与单尺度结构相比, 多尺度设计的典型特征在于下采样模块的引入. ViT[15] 是单尺度直筒型结构的代表, 其在网络不同阶段中使用同等长度或尺寸的图像词符序列; 与之相对应的是以 PVT[22]、Swin Transformer[23] 以及 CrossFromer[52] 等为代表的多尺度金字塔型结构. 多尺度方案可以有效降低网络参数和计算量, 从而使得处理高分辨率数据成为可能. 文献[84]对单尺度和多尺度方法进行了对比, 实验表明多尺度方法相比于单尺度在多种框架中均具备稳定的性能优势.

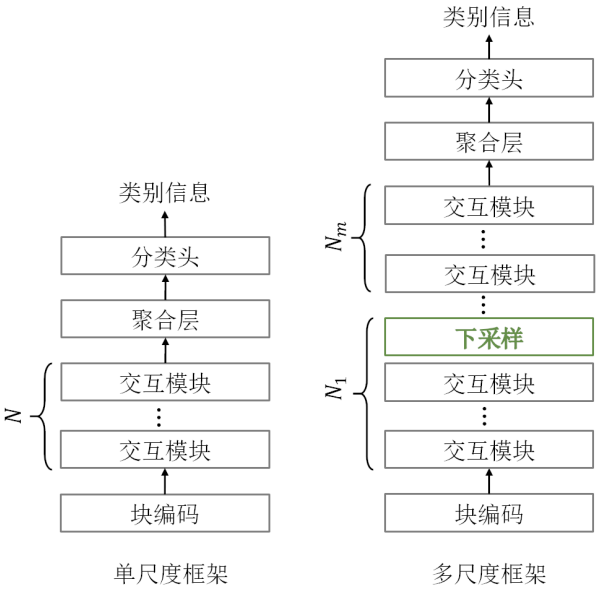


图 4 单尺度与多尺度结构对比

Fig. 4 The comparison of single-scale framework and multi-scale framework

2.4.2 交互模块结构设计

如图 1 所示, 在 ViT[15] 的编码器结构中, 信息交互模块主要由多头注意力层和 MLP 层构成, 其中多头自注意力层主要完成空间层级的信息交互, 而 MLP 主要完成通道级别的信息交互[15]. 当前大多数视觉 Transformer 的交互模块设计基本都遵循了这一范式, 并以自注意力机制为核心. 同多头注意力机制相比, 虽然卷积以及 MLP 在原理和运行机制上与之存在差异, 但它们同样具备空间层级信息交互的能力, 因此许多工作通过引入卷积或 MLP 来替换或增强多头自注意力机制[34,85–91], 形成了多样的交互模块设计方案. 其中最为典型的是以纯 MLP 架构为代表的无自注意力方案[85–88], 和引入

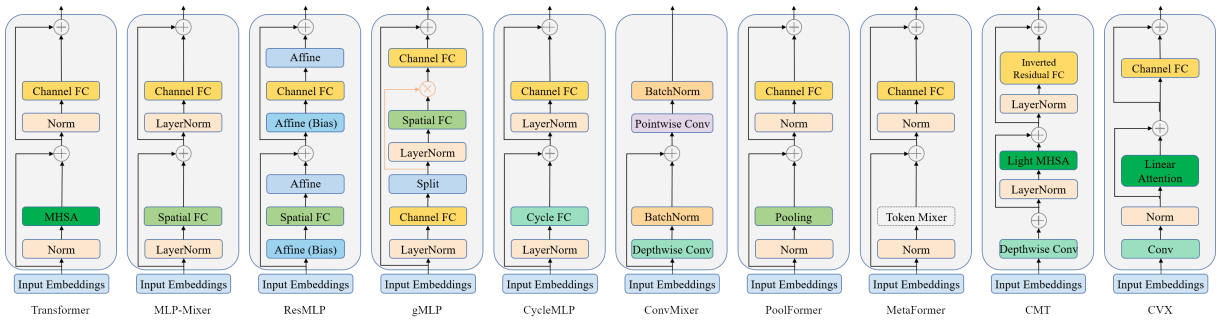


图 5 类 Transformer 方法的交互模块结构对比（Transformer[8], MLP-Mixer[85], ResMLP[86], gMLP[87], CycleMLP[88], ConvMixer[89], PoolFormer[90], MetaFormer[90], CMT[34], CVX[91]）

Fig. 5 The comparison of mixing blocks of Transformer-like methods (Transformer[8], MLP-Mixer[85], ResMLP[86], gMLP[87], CycleMLP[88], ConvMixer[89], PoolFormer[90], MetaFormer[90], CMT[34], CVX[91])

卷积的增强自注意力的方案[34,91]. 为了简洁起见, 在本文后续内容中, 我们将在空间层级进行信息交互的 MLP 称为空间 MLP 机制（Spatial MLP）, 将在通道层级进行信息交互的 MLP 机制称为通道 MLP（Channel MLP）. 不同交互模块的结构如图 5 所示.

1) 无自注意力交互模块: MLP-Mixer[85] 引入了空间 MLP 来替换多头自注意力机制, 成为基于纯 MLP 的类 Transformer 架构的早期代表. 在对图像块序列的特征提取中, MLP-Mixer 在每一层的开始首先将图像块序列转置, 从而实现利用 MLP 进行不同词符之间的交互, 之后经过反转置, 再利用 MLP 进行通道层级的信息交互. 相比于自注意力机制, MLP 的方案实现了类似的词符间信息聚合功能且同样具备全局交互能力; 此外, 由于 MLP 每层的神经元的顺序固定, 因此其具备位置感知能力, 从而不再需要位置编码环节. MLP-Mixer 彻底去除了自注意力机制, 仅依靠纯 MLP 组合取得了与 ViT 相媲美的性能. ResMLP[86] 同样是完全基于 MLP 的架构, 同时其指出纯 MLP 设计相比于基于自注意力的 Transformer 方法在训练稳定性上具备优势, 并提出通过使用简单的仿射变换（Affine transformation）来代替层归一化等规范化方法. gMLP[87] 提出一种基于空间 MLP 的门控机制以替代自注意力, 并使用了通道 MLP—空间门控 MLP—通道 MLP 的组合构建了交互单元. 为了应对 MLP 无法处理变长输入的问题, CycleMLP[88] 提出一种基于循环采样的 MLP 机制, 其在类似卷积核的窗口内部, 按照空间顺序采样该位置的某一通道上的元素, 且不同空间位置的采样元素对应的通道也不同, 从而构建了一种不依赖输入尺寸的空间交互方法, 同时也具备通道交互能力。

基于卷积也可以实现空间信息交互, 从而同样具备取代自注意力的可能, ConvMixer[89] 使用了逐深度卷积（Depthwise convolution）和逐点卷积（Pointwise convolution）来进行空间和通道信息交互, 从而打造了一个基于纯卷积的类 Transformer 网络. PoolFormer[90] 则使用了更为简单的 Pooling 操作来进行空间信息交互, 并进一步提出了更为一般的交互模块方案 MetaFormer[90]. ConNeXt[92] 将 Swin Transformer[23] 网络的特点迁移到卷积神经网络的设计中, 通过调整不同卷积块的比例、卷积核大小、激活函数以及正则化函数等, 使卷积神经网络的结构尽可能趋近 Swin Transformer, 从而在相似计算量下, 实现下超越 Swin Transformer 的性能. RepLKNet[93] 指出在图像处理中, Transformer 的优势可能来源于较大的感受野. 基于这个观点, RepLKNet 通过扩大卷积核, 加入旁路连接和重参数化机制, 来改造卷积神经网络从而取得了媲美 Swin Transformer 的效果.

总的来说, 无论是使用 MLP 还是卷积或者 Pooling 等具备空间交互能力的算子, 在 Transformer 的基本框架下, 替换自注意力模块后依然能够达到与 Transformer 类似的性能. 这也表明, 或许自注意力机制并不是 Transformer 必需的设计, Transformer 的性能可能更多来自于整体的架构[90]以及全局交互给感受野带来的优势[93].

2) 引入卷积的自注意力交互模块: 卷积所具备的局部空间交互性和通道交互性能够有效地与自注意力机制形成互补[84], 通过卷积来增强交互模块的设计在 CMT[34] 以及 CVX[91] 等工作中均进行了尝试并取得了超越基准 Transformer 的效果. 其中 CMT[34] 在自注意力前引入卷积增强局部特性建模, 并在通道 MLP 中加入了卷积增强空间特性建模能力. CVX[91] 使用了 Performer[54] 等线性自注意力机制, 并借助卷积本身的归纳偏置去除了位置编码和类别词符.

表 3 基于 Transformer 和基于 CNN 的目标检测算法在 COCO 2017 val 数据集上的检测精度比较. 其中 C. 表示基于 CNN 的算法, T. 表示基于 Transformer 的算法. 表中数据主要参考文献 [18]

Table 3 The comparison of detection performance of Transformer-based and CNN-based detectors on COCO 2017 val set. C. denotes CNN-based methods, T. denotes Transformer-based methods. Most of the data in the table are from [18]

类型	方法名称	迭代轮次	计算量 (GFLOPs)	参数量 (M)	帧数 (FPS)	多尺度输入	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
C.	FCOS[116]	36	177	-	17	✓	41.0	59.8	44.1	26.2	44.6	52.2
	Faster R-CNN[95]	36	180	42	26	✓	40.2	61.0	43.8	24.2	43.5	52.0
	Faster R-CNN+[95]	108	180	42	26	✓	42.0	62.1	45.5	26.6	45.4	53.4
	Mask R-CNN[99]	36	260	44	-	✓	41.0	61.7	44.9	-	-	-
	Cascade Mask R-CNN[105]	36	739	82	18	✓	46.3	64.3	50.5	-	-	-
T.	ViT-B/16-FRCNN‡[117]	21	-	-	-	-	36.6	56.3	39.3	17.4	40.0	55.5
	ViT-B/16-FRCNN*[117]	21	-	-	-	-	37.8	57.4	40.1	17.8	41.4	57.3
	DETR-R50[16]	500	86	41	28	-	42.0	62.4	44.2	20.5	45.8	61.1
	DETR-DC5-R50[16]	500	187	41	12	-	43.3	63.1	45.9	22.5	47.3	61.1
	ACT-MTKD (L=16)[113]	-	156	-	14	-	40.6	-	-	18.5	44.3	59.7
	ACT-MTKD (L=32)[113]	-	169	-	16	-	43.1	-	-	22.2	47.1	61.4
	Deformable DETR[24]	50	78	34	27	-	39.7	60.1	42.4	21.2	44.3	56.0
	Deformable DETR-DC5[24]	50	128	34	22	-	41.5	61.8	44.9	24.1	45.3	56.0
	Deformable DETR[24]	50	173	40	19	✓	43.8	62.6	47.7	26.4	47.1	58.0
	Two-Stage Deformable DETR[24]	50	173	40	19	✓	46.2	65.2	50.0	28.8	49.2	61.7
	SMCA[110]	50	86	40	22	-	41.0	-	-	21.9	44.3	59.1
	SMCA+[110]	108	86	40	22	-	42.7	-	-	22.8	46.1	60.0
	SMCA[110]	50	152	40	10	✓	43.7	63.6	47.2	24.2	47.0	60.4
	SMCA+[110]	108	152	40	10	✓	45.6	65.5	49.1	25.9	49.3	62.6
	Efficient DETR[109]	36	159	32	-	✓	44.2	62.2	48.0	28.4	47.5	56.6
	Efficient DETR*[109]	36	210	35	-	✓	45.1	63.1	49.1	28.3	48.4	59.0
	Conditional DETR[111]	108	90	44	-	-	43.0	64.0	45.7	22.7	46.7	61.5
	Conditional DETR-DC5[111]	108	195	44	-	-	45.1	65.4	48.5	25.3	49.0	62.2
	UP-DETR[112]	150	86	41	28	-	40.5	60.8	42.6	19.0	44.4	60.0
	UP-DETR+[112]	300	86	41	28	-	42.8	63.0	45.3	20.8	47.1	61.7
	TSP-FCOS[115]	36	189	51.5	15	✓	43.1	62.3	47.0	26.6	46.8	55.9
	TSP-RCNN[115]	36	188	64	11	✓	43.8	63.3	48.3	28.6	46.9	55.7
	TSP-RCNN+[115]	96	188	64	11	✓	45.0	64.5	49.6	29.7	47.7	58.0
	YOLOS-S[114]	150	200	30.7	7	-	36.1	56.4	37.1	15.3	38.5	56.1
	YOLOS-S[114]	150	179	27.9	5	✓	37.6	57.6	39.2	15.9	40.2	57.3
	YOLOS-B[114]	150	537	127	-	-	42.0	62.2	44.5	19.5	45.3	62.1

3 视觉 Transformer 的一般性框架

视觉 Transformer 结构的设计是一个活跃的研究方向, 无论是 ViT[15] 还是后续的改进方法, 都很好地拓展了视觉 Transformer 的设计思路. 但目前仍然缺乏对视觉 Transformer 通用设计方案的讨论. 本节以底层视觉分类任务为例, 给出视觉 Transformer 的一般性框架 VTA(Vision Transformers Architecture), 如图 6 所示。VTA 给出的视觉 Transformer 一般性框架包含七层: 输入层、序列化层、位置编码层、交互层、采样层、聚合层以及输出层. 其中输入层和输出层分别完成对输入的读取和结果的产生, 下面将对剩余各层进行简要介绍。

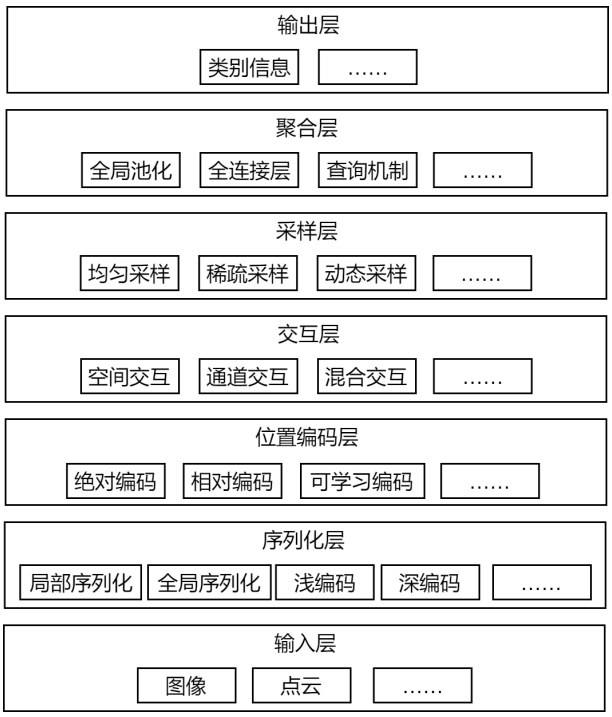


图 6 视觉 Transformer 的一般性框架

Fig. 6 Vision Transformers architecture

3.1 序列化层

序列化层的功能在于将输入划分为词符序列的形式, 并进行序列编码. 其中, 序列划分方式可以分为局部序列划分和全局序列划分. 局部序列划分将序列分组, 位于同一组的词符可在后续环节进行交互, 典型的局部序列划分方法有 Swin Transformer[23] 等. 全局序列划分则是更一般的序列划分方法, 这种方式下, 全部词符均可以进行直接交互. 对编码方式而言, 主要有浅编码和深编码两种方式, 相对于浅编码方案, 深度编码利用更多的卷积层对图像或划分后的序列进行处理, 更有利于视觉 Transformer 的训练和优化[73].

3.2 位置编码层

对不具备位置感知能力的视觉 Transformer 方案, 位置编码层被用来显式地提取位置信息. 位置编码方案主要包括绝对位置编码、相对位置编码以及可学习位置编码. 绝对位置编码仅考虑词符在序列中的位置信息, 相对位置编码则考虑词符对之间的相对位置信息, 更有利于提高模型的表达能力[78]. 此外, 位置编码还可以可学习的方式进行[16], 以建立更为一般的位置编码信息.

3.3 交互层

交互层旨在对词符序列中的特征进行交互, 主要可分为空间交互、通道交互和混合交互模式. 原始的 Transformer 方案[15] 将空间交互和通道交互分离, 并使用基于自注意力机制实现空间交互功能. 其通过计算词符对之间的相似性来进行加权信息聚合. 基于注意力机制的空间交互是早期 Transformer 方法的典型特质. 但随着更多相关工作的开展, 研究人员发现, 自注意力机制也仅是空间交互功能的一种实现方式, 其可以被卷积或空间 MLP 所替代. 通道信息交互常用的方法是通道 MLP. 混合交互机制则打破了空间和通道独立的限制, 利用包括卷积在内的算子, 同时建立词符在空间和通道中的关系[73,89,90].

3.4 采样层

采样层旨在对词符序列进行采样或合并, 以减少序列中词符个数, 从而降低计算量. 常见的采样方式包括均匀采样、稀疏采样以及动态采样. 其中, 均匀采样[22] 通过池化层或卷积层对相邻词符进行合并; 稀疏采样[24,52] 则在更大的范围内进行词符的选择或合并, 有利于提高感受野范围. 动态采样[51] 是一种更为一般性的采样方案, 其往往通过可学习的过程从输入的词符序列中自适应地选择一些数量的词符, 作为后续网络的输入.

3.5 聚合层

对分类任务而言, 聚合层主要完成对词符特征全局信息的聚合. 全局池化、全连接层是常见的全局信息聚合方式. 这两种方式都属于静态聚合方案, 其聚合方式不随输入内容变化而改变. ViT[15] 使用了基于类别词符的查询机制, 通过定义可学习和更新的类别词符变量, 并与输入词符序列进行互注意力实现对信息的动态聚合.

4 基于 Transformer 的目标检测模型

基于卷积神经网络的目标检测模型训练流程主要由特征表示, 区域估计和真值匹配三部分组成:

1) 特征表示: 特征表示基于卷积神经网络来提取输入的语义特征[5,94].

2) 区域估计: 区域估计通过区域特征提取算子, 如卷积、裁剪、感兴趣区域池化 (RoI pooling)[95] 或感兴趣区域对齐 (RoI align)[99] 等, 获得局部特征, 并对局部输入的类别和位置等信息进行估计和优化.

3) 真值匹配: 基于卷积神经网络的真值匹配往往通过具备位置先验的匹配策略, 如重叠度 (IoU)、距离等, 进行标注框同锚点框[95,100]、关键点[101] 或中心点[102] 等参考信息之间的匹配, 建立参考信息的真值, 以此作为网络学习的监督信息.

基于 Transformer 的目标检测模型拓展了以上三个过程的实现方式. 在特征学习方面, 基于 Tranformer 的特征构建方式可以取代卷积神经网络的角色[23]; 在区域估计方面, 基于编码器—解码器的区域估计方式也被大量尝试和验证[16]; 在真值匹配方面, DETR[16] 提出了基于二分匹配 (Bipartite matching) 的真值分配方式, 该方法事先不依赖于位置先验信息, 而是将预测结果产生后将预测值同真实值进行匹配. 本节将从以上三个角度对基于 Transformer 的工作进行介绍. 表 3 总结了不同基于 Transformer 的目标检测模型在 COCO[103] 数据集上的性能对比.

4.1 利用 Transformer 进行目标检测网络的特征学习

作为特征提取器, Transformer 网络具有比 CNN 更大的感受野和更灵活的表达方式, 因此也有望取得更好的性能以为下游任务提供高质量输入. 考虑到特征学习属于 Transformer 网络的基础功能, 并已在第 2 节中进行了详细梳理, 因此本节将简要介绍其设计, 而更多地关注此类方法在目标检测器中的应用.

基于层级结构设计的 PVT[22]、基于卷积和 Transformer 融合的 CMT[34]、基于局部—整体交互的 Cross Former[52]、Conformer[76] 以及基于局部窗口设计的 Swin Transformer[23] 均被成功应用到了 RetinaNet[104]、Mask R-CNN[99]、Cascade R-CNN[105]、ATSS[106]、RepPoints-v2[107] 和 Sparse RCNN[108] 等典型目标检测网络中, 相比于 ResNet 等卷积神经网络取得了更好的效果. 这类方法基于典型的目标检测流程, 将 Transformer 作为一种新的特征学习器, 替代原有的卷积神经网络骨架, 从而完成目标检测任务.

4.2 利用 Transformer 进行目标信息估计

不同于 CNN 利用卷积实现对区域信息的估计和预测, 基于 Transformer 的目标检测网络使用了查询机制, 通过查询与特征图的注意力交互实现对目标位置、类别等信息的估计. 本小节将以 DETR[16] 中的目标查询机制为例介绍查询机制的作用, 并总结目前存在的问题以及解决方案. DETR 的基本结构如图 7 所示.

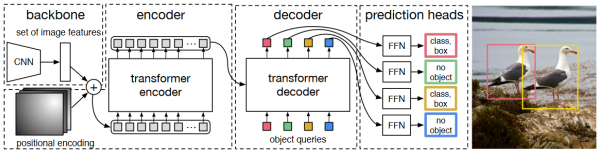


图 7 DETR 的结构图[16]

Fig. 7 The framework of DETR[16]

4.2.1 DETR 中的目标查询机制

DETR[16] 首先通过编码器提取图像特征, 之后利用随机初始化的目标查询机制来与图像特征进行交互, 以互注意力的机制进行目标级别信息的提取, 经过多层交互之后, 利用全连接层从每个目标查询中预测目标的信息, 形成检测结果.

目标查询向量包含了潜在目标的位置信息和特征信息, 其与图像特征进行交互的过程实现了从全局信息中对潜在目标特征的抽取, 同时完成了对预测位置的更新. 多个查询层的堆叠构建了一种类似 Cascade RCNN[105] 的迭代网络[109], 以更新目标查询的方式实现对位置和特征信息的优化. 为了清楚地介绍 Transformer 的设计机制, 本文将目标查询所表示的内容分成两部分, 一部分是与特征内容有关的, 记为内容嵌入 (Content embedding), 一部分是与位置有关的, 记作位置嵌入 (Positional embedding).

这种目标查询的方式实现了较为有效的目标检测功能, 但同时存在着收敛速度较慢[24,110,111] (DETR 需要 500 个轮次的训练才能收敛)、小目标检测效果不佳[24] 以及查询存在冗余[113] 等问题. 其中, 针对小目标检测效果差的问题, 现有文献的主要做法是利用多尺度特征[24], 通过在不同分辨率特征图上进行目标查询, 增加对小目标物体的信息表示, 以提高小目标的准确率. 针对目标查询存在冗余的现象, ACT[113] 提出使用局部性敏感哈希 (Locality sensitivity hashing, LSH) 算法实现自适应聚类, 以压缩目标查询的个数, 从而实现更为高效的目标查询. 本小节将主要针对以 DETR[16] 为代表的网络收敛速度慢的问题, 分析其原因并总结提升训练速度的方案.

4.2.2 收敛速度提升

图 8 展示了 DETR[16] 以及其改进方法与基于 CNN 的检测器的收敛速度对比, 可以看到 DETR 需要长达 500 个轮次的训练才能得到较为稳定的效果. 其收敛较慢的主要原因在于目标查询机制的设计[24,110,111], 本节从查询初始化、参考点估计和目标分布三个方面分析 DETR 的设计并总结了提升收敛速度的方法.

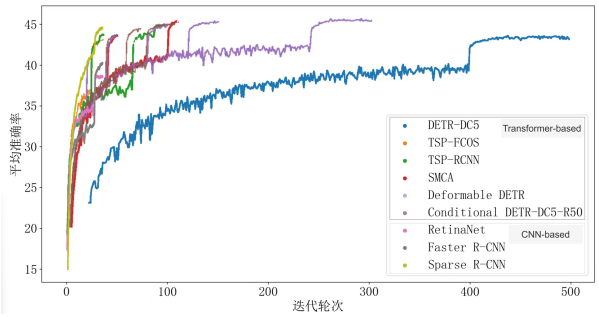


图 8 基于 Transformer 和 CNN 的目标检测器的收敛速度对比 (基于 Transformer 的: DETR-DC5[16], TSP-FCOS[115], TSP-RCNN[115], SMCA[110], Deformable DETR[24], Conditional DETR-DC5-R50[111]; 基于 CNN 的: RetinaNet[104], Faster R-CNN[95], Sparse R-CNN[108])

Fig. 8 The comparison of converge speed among object detectors based on Transformer and CNN(Transformer-based: DETR-DC5[16], TSP-FCOS[115], TSP-RCNN[115], SMCA[110], Deformable DETR[24], Conditional DETR-DC5-R50[111]. CNN-based: RetinaNet[104], Faster R-CNN[95], Sparse R-CNN[108])

1) 输入依赖的目标查询初始化: DETR[16] 对目标查询使用了随机初始化的方法, 通过训练时的梯度更新来实现对目标查询输入的优化, 以学习输入数据集中的物体的统计分布规律. 这种方式需要较长的过程才能实现对物体物质分布的学习, 其可视化表现为交叉注意图 (Cross-attention map) 的稀疏程度需要较长的训练轮次才能收敛[115] (如图 9 所示). 此外, 关于目标分布的统计信息属于一种数据集层面的特征, 无法实现对具体输入的针对性初始化, 也影响了模型的收敛速度.

为了改善由于初始化而造成的收敛问题, TSP[115], Efficient DETR[109] 等工作提出了输入依赖的查询初始化方法, 从输入图像特征中预测潜在目标的位置和尺寸等信息, 作为初始的目标查询输入到编码器或解码器网络, 进而得到最终的目标检测结果. 其中, TSP[115] 使用了 CNN 网络作为产生初始目标查询的途径, 借鉴 FCOS[116] 和 RCNN[118] 的思路, 分别提出了 TSP-FCOS 和 TSP-RCNN 进行图像中目标信息的估计, 并借助 Transformer 编码器实现对目标估计的优化; Efficient DETR[109] 使用基于 Transformer 的编码器网络学习到的词符特征进行密集预测, 得到相应位置可能的目标的位置、尺寸和类别信息, 并选择置信度较高的结果作为目标查询的初始状态, 然后利用解码器进行稀疏预测, 得到最终结果.

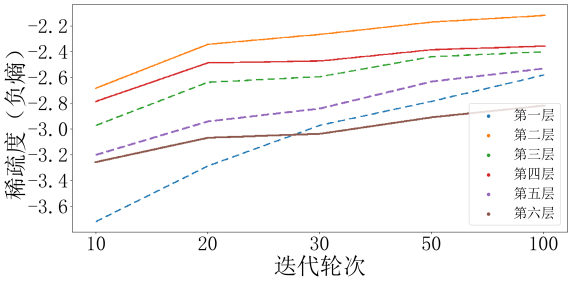


图 9 DETR 交叉注意力稀疏性变化

Fig. 9 The change of sparsity of cross-attention map in DETR

TSP[115] 和 Efficient DETR[109] 所提出的目标查询初始化方法一方面能够根据不同输入得到不同的目标查询初始化结果, 是一种输入依赖的初始化方式; 另一方面, 实现了目标查询所包含的内容嵌入和位置嵌入的显式对齐, 从而较好地加速了目标检测器的收敛.

2) 输入依赖的位置嵌入更新: DETR 位置嵌入的弱定位能力也是影响 DETR 模型收敛的主要原因之一. 在 DETR[16] 解码器中的多层网络中, 目标查询的内容嵌入通过交叉注意力实现对自身信息的更新, 但位置嵌入并不在层之间进行更新. 这种方式一方面导致了位置嵌入与内容嵌入的不匹配, 另一方面还导致位置嵌入难以准确表达潜在目标的准确位置信息, 使得获取位置信息的任务转移到内容嵌入中[111]. Conditional DETR[111] 通过对比实验发现, 去掉解码器中第二层之后的位置嵌入信息, DETR 的平均准确率仅下降 0.9%, 从而说明了在原始的 DETR 的解码器中的位置嵌入所发挥的作用很小.

Deformable DETR[24]、SMCA[110] 和 Conditional DETR[111] 等方法从每层输入信息中学习位置嵌入信息的更新, 能够较好地弥补 DETR 设计中位置嵌入定位能力不足的缺陷. 其中,Deformable DETR[24] 和 SMCA[110] 从目标查询中预测了每个查询对应的参考点坐标, 来提高定位能力; Conditional DETR[111] 利用目标查询预测二维坐标信息, 并利用内容嵌入学习对坐标嵌入信息的变换, 使位置嵌入和内容嵌入在统一空间, 进而使得目标查询和键值在统一空间, 从而提高相似性判别和定位能力.

表 4 基于 Transformer 的语义分割算法在 ADE20K val 数据集上的语义分割精度比较. 其中, 1k 表示 ImageNet-1k, 22k 表示 ImageNet-1k 和 ImageNet-21k 的结合. 表中数据主要参考文献 [18]

Table 4 The comparison of semantic segmentation performance of Transformer-based methods on ADE20K val set. 1k denotes ImageNet-1k dataset, and 22k denotes the combination of ImageNet-1k and ImageNet-21k. Most of the data in the table are from [18]

方法名称	骨干网络	预训练数据集	图像尺寸	参数量 (M)	计算量 (GFLOPs)	帧数 (FPS)	多尺度输入	mIoU
UperNet [122]	R-50	1k	512	-	-	23.4	✓	42.8
	R-101	1k	512	86	257	20.3	✓	44.9
	Swin-T	1k	512	60	236	18.5	✓	46.1
	Swin-S	1k	512	81	259	15.2	✓	49.3
	Swin-B	22k	640	121	471	8.7	✓	51.6
	Swin-L	22k	640	234	647	6.2	✓	53.5
Segformer [25]	MiT-B3	1k	512	47.3	79	-	✓	50.0
	MiT-B4	1k	512	64.1	95.7	15.4	✓	51.1
	MiT-B5	1k	640	84.7	183.3	9.8	✓	51.8
Segmenter [124]	ViT-S/16	-	512	27	-	34.8	✓	46.9
	ViT-B/16	-	512	106	-	24.1	✓	50.0
	ViT-L/16	-	640	334	-	-	✓	53.6
MaskFormer [125]	R-50	1k	512	41	53	24.5	✓	46.7
	R-101	1k	512	60	73	19.5	✓	47.2
	Swin-T	1k	512	42	55	22.1	✓	48.8
	Swin-S	1k	512	63	79	19.6	✓	51.0
	Swin-B	22k	640	102	195	12.6	✓	53.9
	Swin-L	22k	640	212	375	7.9	✓	55.6
Mask2Former [26]	R-50	1k	512	-	-	-	✓	49.2
	R-101	1k	512	-	-	-	✓	50.1
	Swin-S	1k	512	-	-	-	✓	52.4
	Swin-B	22k	640			-	✓	55.1
	Swin-L	22k	640	-	-	-	✓	57.3

表 5 基于 Transformer 的实例分割方法和基于 CNN 算法在 COCO test-dev 数据集上的实例分割精度比较. 表中数据主要参考文献 [18]

Table 5 The comparison of instance segmentation performance of Transformer-based and typical CNN-based methods on COCO test-dev dataset. Most of the data in the table are from [18]

方法名称	骨干网络	迭代轮次	帧数 (FPS)	Ap^m	Ap^m_S	Ap^m_M	Ap^m_L	Ap^b
Mask R-CNN [99]	R-50-FPN	36	15.3	37.5	21.1	39.6	48.3	41.3
	R-101-FPN	36	11.8	38.8	21.8	41.4	50.5	43.1
Blend Mask [96]	R-50-FPN	36	15.0	37.8	18.8	40.9	53.6	43.0
	R-101-FPN	36	11.5	39.6	22.4	42.2	51.4	44.7
SOLO v2 [97]	R-50-FPN	36	18.0	38.8	16.5	41.7	56.2	40.4
	R-101-FPN	36	9.0	39.7	17.3	42.9	57.4	42.6
ISTR [127]	R-50-FPN	36	13.8	38.6	22.1	40.4	50.6	46.8
	R-101-FPN	36	11.0	39.9	22.8	41.9	52.3	48.1
SOLQ [98]	R-50	50	-	39.7	21.5	42.5	53.1	47.8
	R-101	50	-	40.9	22.5	43.8	54.6	48.7
	Swin-L	50	-	45.9	27.8	49.3	60.5	55.4
QueryInst [126]	R-50-FPN	36	7.0	40.6	23.4	42.5	52.8	45.6
	R-101-FPN	36	6.1	41.7	24.2	43.9	53.9	47.0
	Swin-L	50	3.3	49.1	31.5	51.8	63.2	56.1
Mask2Former [26]	Swin-L	-	-	50.5	29.1	53.8	71.2	-

表 6 基于 Transformer 的全景分割算法在 COCO panoptic minval 数据集上的全景分割精度比较. 表中数据主要参考文献 [18]

Table 6 The comparison of panoptic segmentation performance of Transformer-based methods on COCO panoptic minival dataset. Most of the data in the table are from [18]

方法名称	骨干网络	迭代轮次	参数量 (M)	计算量 (GFLOPs)	PQ	PQ^{Th}	PQ^{St}
DETR [16]	R-50	500+25	42.8	137	43.4	48.2	36.3
	R-101		61.8	157	45.1	50.5	37
MaxDeepLab [123]	Max-S	54	61.9	162	48.4	53.0	41.5
	Max-L		451	1 846	51.1	57.0	42.2
MaskFormer [125]	R-50	300	45	181	46.5	51.0	39.8
	R-101		64	248	47.6	52.5	40.3
	Swin-T		42	179	47.7	51.7	41.7
	Swin-S		63	259	49.7	54.4	42.6
	Swin-B		102	411	51.1	56.3	43.2
	Swin-L		212	792	52.7	58.5	44.0
Panoptic SegFormer [128]	R-50	12	51.0	214	48.0	52.3	41.5
	R-50	24	51.0	214	49.6	54.4	42.4
	R-101		69.9	286	50.6	55.5	43.2
	Swin-L		221.4	816	55.8	61.7	46.9

3) 显式目标分布建模: DETR[16] 使用了信息相似性度量来实现在全局范围内的目标嵌入的信息聚合, 这种方式有助于更完全地获取目标的信息, 但同时也可能引入较多的噪声干扰[24], 从而影响学习过程, 而且, 从全局信息中收敛到潜在目标的局部空间也需要较长的训练过程.

建立对潜在目标分布空间的建模机制有助于加速目标检测过程, 减少训练时间, 同时减少噪声的引入[24,110]. 矩形分布假设是基于 CNN 的目标检测器的常用设计之一[95,100], 在基于 Transformer 的目标检测器中, 虽然图片以序列的方式进行编码和解码, 但仍可以借助逆序列化获取二维的图片结构的数据. 并在此基础上, 实现类似 CNN 网络中的感兴趣区域池化等操作, 以此实现对目标空间的建模. 现有对 Transformer 目标分布进行显式建模的方法主要有两种: 散点分布[24] 和高斯分布[110].

散点分布: Deformable DETR[24] 利用了散点采样实现对目标空间分布的建模. 针对每一个目标查询, Deformable DETR 首先从中学习目标的参考点坐标、采样点坐标和采样点权重, 然后在若干采样点之间计算局部范围内的注意力, 并进行信息聚合. 这种方式大大减少了计算量, 同时可以较灵活地模拟目标的空间分布, 实现对于与目标查询有关联的点的聚合, 从而加速了网络的收敛过程.

高斯分布: SMCA[110] 提出了一种利用高斯函数建模目标空间分布, 实现局部信息聚合的方法. SMCA 首先从目标查询中学习潜在目标的位置和尺寸信息, 之后, 根据预测得到的位置和物体尺寸信息建立二维高斯分布函数, 来对近距离特征赋予较高权重, 对远距离特征赋予较低权重.

4.3 基于 Transformer 的目标检测模型的真值匹配

DETR[16] 将目标检测建模为集合预测的问题, 并使用了二分匹配 (Bipartite Matching) 来为目标查询赋予对应的真值. 二分匹配利用匈牙利算法来进行快速实现. 定义: σ 表示匹配策略, $y_i = (c_i, b_i)$ 表示真实值, c_i 表示真实类别, b_i 表示标注框的值, $\hat{y}_{\sigma(i)} = (\hat{p}_{\sigma(i)}, \hat{b}_{\sigma(i)})$ 表示第 $\sigma(i)$ 个预测值. 则 y_i 与 $\hat{y}_{\sigma(i)}$ 的匹配损失为:

$$\mathcal{L}_{\text{match}}\left(y_i, \hat{y}_{\sigma(i)}\right) = -\mathbb{1}_{\{c_i \neq \varnothing\}} \hat{p}_{\sigma(i)}\left(c_i\right) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}\left(b_i, \hat{b}_{\sigma(i)}\right) \tag{32}$$

最佳匹配定义为:

$$\hat{\sigma} = \arg\min_{\sigma \in \mathfrak{S}_N} \sum_{i}^{N} \mathcal{L}_{\text{match}}\left(y_i, \hat{y}_{\sigma(i)}\right) \tag{33}$$

不同于 CNN 中基于锚点框或关键点的真值匹配方式, 二分匹配是在得到预测结果后进行, 基本上是一种不确定性策略, 且容易受到训练过程的干扰[115], 进而导致训练过程 (尤其是训练过程的早期) 收敛速度较慢. 针对这个问题 TSP[115] 基于 FCOS 提出了一种新的匹配策略, 仅将落在真实标注框内的预测值或与标注框有一定重合的预测值与该真实值进行匹配, 从而加速收敛速度.

5 基于 Transformer 的图像分割模型

图像分割主要包括语义分割, 实例分割和全景分割[119], 近些年, 以 FCN[120]、DeepLab[45]、Mask RCNN[99] 等方法为代表的图像分割方法已经取得了较好的效果, 但这种基于卷积神经网络的图像分割方法在建立远程依赖上依旧存在不足. 相比之下, Transformer 网络所具备的全局信息交互能力能够帮助特征提取器快速建立全局感受野, 从而实现更准确的场景理解[121]. 表 4 、表 5 和表 6 分别展示了基于 Transformer 的语义分割、实例分割和全景分割方法的结果以及其与经典 CNN 方法的对比. 本节将主要从特征提取、分割结果生成两个方面介绍 Transformer 在图像分割中的应用.

5.1 基于 Transformer 进行分割网络的特征学习

Transformer 网络以一定尺寸的图像块作为最小特征单元, 其编码后的特征经过上采样操作就可以集成到现有的图像分割框架中. Transformer 以其全局感受野和动态交互能力, 使得图像分割模型能够对图像中的上下文关系进行充分表示和建模, 从而取得更好的效果[22,23,34,52,53,72,76].

除了将 Transformer 集成到现有分割框架以替换 CNN 之外, 近期的一些工作还针对 Transformer 设计了新的分割框架以充分利用其在有效感受野等方面的优势[25,121]. 其中, SETR[121] 以序列学习的视角提出了基于 ViT[15] 的完全由自注意力机制构成的特征编码网络, 并在此基础上提出了三种解码方案 (简单上采样解码器、渐进式解码器和多尺度融合解码器) 产生分割结果, 打破了语义分割任务基于编码器—解码器的 FCN 范式, 其结构如图 10 所示. SegFormer[25] 针对 SETR 柱状编码方式计算量较大以及固定位置编码不利于拓展等问题, 提出了使用具备层次结构的 Transformer 网络以保留粗粒度和细粒度两种特征, 并通过在自注意力中引入卷积机制来去除位置编码提高了网络灵活性. SegFormer 同时指出, 基于 Transformer 的图像分割网络可以在仅适用较为简单的解码器的情况下, 实现不错的效果, 并提出了一种仅包含数个线性层的解码器方案.

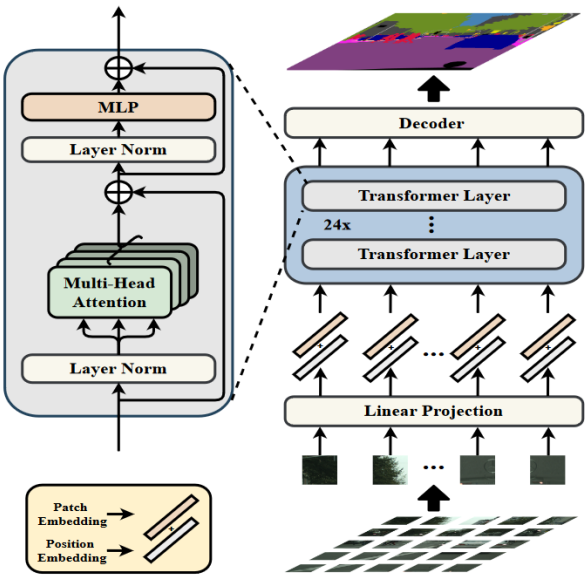


图 10 SETR 的结构图[121]

Fig. 10 The framework of SETR[121]

5.2 利用 Transformer 产生图像分割结果

像素分割和实例分割是图像分割中的两个基本任务, 在基于卷积神经网络的方法中, 前者往往基于编码器—解码器的结构产生, 后者则通常借助 RCNN 实现对目标级别信息的输出[119]. Transformer 的出现, 尤其是其目标查询机制, 为解决图像分割提供了一种新的思路, 而且有望以一种统一的方式实现像素和实例级别的分割. Transformer 的查询机制可以用来表示多种信息, 既可以表示类别信息[124]、位置信息[26,125] 同时也可以表示其它特征信息[126], 这种具备通用性的表示形式为实现统一形式的图像分割提供了基础. 本节将主要从基于目标查询的语义分割和实例分割两方面介绍 Transformer 给图像分割领域带来的启发和改变, 并结合全景分割, 总结以统一的方式进行图像分割的工作.

5.2.1 基于查询的语义分割方法

按照产生结果的形式, 基于查询的语义分割方法可以分为像素级预测[124] 和掩码级预测[125], 前者为每一个像素输出一个类别信息, 后者则对掩码内的像素统一预测一个类别信息. 在语义分割任务中, 查询以随机初始化的方式产生, 之后通过与图像特征的交互实现对类别信息的提取, 并最终用于产生分割结果.

1) 像素级语义分割: Segmenter[124] 利用类别嵌入 (Class embedding) 建立目标查询, 通过交叉注意力与图像序列进行信息交互, 最终利用类别嵌入与图像序列之间的注意力图进行图像块的逐像素分割结果预测.

2) 掩码级语义分割: MaskFormer[125] 借鉴了 DETR[16] 中的集合预测思想, 提出了掩码级的语义分割思路, 其使用了 Transformer 和 CNN 两种解码器, 其中 Transformer 解码器基于随机初始化的查询实现对类别信息的预测, CNN 解码器则通过常规卷积实现对二进制掩码信息的预测, 最后通过融合类别预测和掩码预测得到语义分割结果. 这种掩码级的语义分割结果生成方式一方面简化了语义分割任务, 另一方面能够与实例分割实现较好的统一. 在性能上, MaskFormer 也验证了在类别数目较多的情况下, 基于掩码的语义分割相比于像素级分割方式在性能上更具优势.

Mask2Former[26] 进一步提升了掩码级语义分割的性能和训练速度, 其基于 MaskFormer[125] 提出了利用多尺度特征来增强对小目标的分割能力, 同时使用了掩码注意力来关注目标局部信息, 从而加速 Transformer 网络的收敛速度.

5.2.2 基于目标查询的实例分割和全景分割方法

在基于 Transformer 的语义分割方法中[26,125], 查询通常与类别信息相关, 而在实例分割中, 查询则往往与前景目标的位置和特征相关[126–128], 这与基于 Transformer 的目标检测网络中的查询机制所表示的信息基本一致[16]. 根据目标信息预测和掩码生成的顺序, 本小节将基于目标查询的实例 / 全景分割方法分为基于检测的分割方法和检测分割并行的方法.

1) 基于检测的实例 / 全景分割方法: DETR[16] 在目标检测结果的基础上生成检测框嵌入, 通过与图像编码特征进行交互提取目标特征, 之后基于查询与图像特征的注意力图进行目标和背景掩码的预测. 不同于 DETR[16] 中将目标和背景均表示为检测框的方式, Panoptic SegFormer[128] 提出区分前景目标和背景信息更有利于产生准确的背景预测. 在解码阶段, Panoptic SegFormer 首先使用位置解码器针对前景目标提取目标信息, 在此基础上引入背景查询, 并利用掩码解码器产生掩码结果.

2) 检测分割并行的实例分割方法: 基于 Sparse RCNN[108], QueryInst[126] 和 ISTR[127] 提出了检测分割并行的实例分割方法. 其中, QueryInst[126] 基于随机初始化的目标位置从图像中获取区域信息, 同时以随机初始化的方式生成目标特征信息, 之后通过不断地迭代, 优化查询的学习以及对目标的信息提取. 目标特征信息用于学习动态卷积的参数以实现对区域特征的动态处理, 在此基础上并行产生包围框和掩码预测. ISTR[127] 同样采用了随机初始化的查询来表示目标的包围框信息, 但采用了图像特征作为产生动态卷积参数的输入. QueryInst[126]

和 ISTR[127] 这种基于查询的迭代式预测方式降低了对目标包围框预测的要求, 使得随机初始化的目标信息依然能够在几轮迭代之后建立对目标的准确描述.

6 总结与展望

本文介绍了视觉 Transformer 模型基本原理和结构, 以图像分类为切入点总结了 Transformer 作为骨干网络的关键研究问题和最新进展, 并提出了视觉 Transformer 的一般性框架, 同时以目标检测和图像分割为例介绍了视觉 Transformer 模型在上层视觉任务中的应用情况. 视觉 Transformer 网络作为一种新的视觉特征学习网络, 在连接范围、权重动态性以及位置表示能力等方面与 CNN 网络有着较大的差异. 其远距离建模能力和动态的响应特质使之具备了更为强大的特征学习能力, 但同时也带来了严重的数据依赖和算力资源依赖等问题. 对视觉 Transformer 的效率和能力的研究仍将是未来的主要研究方向之一, 此外, Transformer 模型为多模态数据特征学习和多任务处理提供了一种统一的解决思路, 基于 Transformer 的视觉模型有望实现更好的信息融合和任务融合.

References

1 ZHANG Hui, WANG Kun-Feng, WANG Fei-Yue. Advances and Perspectives on Applications of Deep Learning in Visual Object Detection. ACTA AUTOMATICA SINICA, 2017, 43(8): 1289-1305. doi: 10.16383/j.aas.2017.c160822
(张慧, 王坤峰, 王飞跃. 深度学习在目标视觉检测中的应用进展与展望. 自动化学报, 2017, 43(8): 1289-1305. doi: 10.16383/j.aas.2017.c160822)

2 CHEN Wei-Hong, AN Ji-Yao, LI Ren-Fa, LI Wan-Li. Review on Deep-learning-based Cognitive Computing. ACTA AUTOMATICA SINICA, 2017, 43(11): 1886-1897. doi: 10.16383/j.aas.2017.c160690
(陈伟宏, 安吉尧, 李仁发, 李万里. 深度学习认知计算综述. 自动化学报, 2017, 43(11): 1886-1897. doi: 10.16383/j.aas.2017.c160690)

3 LeCun Y, Boser B, Denker J S, Henderson D, H Richard E, Hubbard W, Jackel L D. Backpropagation applied to handwritten zip code recognition[J]. Neural computation, 1989, 1(4): 541-551.

4 Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25: 1097-1105.

5 He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

6 Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

7 Chung J, Gulcehre C, Cho K H, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. arXiv preprint arXiv:1412.3555, 2014.

8 Vaswani A, Shazeer N, Parmar N, Uszkoreit U, Jones L, Gomez A, Kaiser L, Polosukhin I. Attention is all you need[C]//Advances in neural information processing systems. 2017: 5998-6008.

9 Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. ICLR, 2014.

10 Gehring J, Auli M, Grangier D, Yarats D, Dauphin Y N. Convolutional sequence to sequence learning[C]//International Conference on Machine Learning. PMLR, 2017: 1243-1252.

11 Jozefowicz R, Vinyals O, Schuster M, Shazeer N, Wu Y. Exploring the limits of language modeling[J]. arXiv preprint arXiv:1602.02410, 2016.

12 Luong M T, Pham H, Manning C D. Effective approaches to attention-based neural machine translation[J]. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015.

13 Devlin J, Chang M W, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2018.

14 Brown T B, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler D M, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D. Language models are few-shot learners[J]. Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems.14165, 2020.

15 Dosovitskiy A, Beyer L, Kolesnikov A, Dirk W, Xiaohua Z, Thomas U, Mostafa D, Matthias M, Georg H, Sylvain G, Jakob U, Neil H. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.

16 Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-end object detection with transformers[C]//European Conference on Computer Vision. Springer, Cham, 2020: 213-229.

17 Han K, Wang Y, Chen H, Chen X, Guo J, Liu Z, Tang Y, Xiao A, Xu C, Xu Y. A survey on visual transformer[J]. arXiv preprint arXiv:2012.12556, 2020.

18 Liu Y, Zhang Y, Wang Y, Hou F, Yuan J, Tian J, Zhang Y, Shi Z, Fan J, He Z. A Survey of Visual Transformers[J]. arXiv preprint arXiv:2111.06091, 2021.

19 Khan S, Naseer M, Hayat M, Zamir S W, Khan, F S, Shah M. Transformers in vision: A survey[J]. arXiv preprint arXiv:2101.01169, 2021.

20 Selva J, Johansen A S, Escalera S, Nasrollahi K, Moeslund T B, Clapés A. Video Transformers: A Survey[J]. arXiv preprint arXiv:2201.05991, 2022.

21 Shamshad F, Khan S, Zamir S W, Khan M H, Hayat M, Khan F S, Fu H. Transformers in Medical Imaging: A Survey[J]. arXiv preprint arXiv:2201.09873, 2022.

22 Wang W, Xie E, Li X, Fan Deng-Ping, Song K, Liang D Lu T, Lou P, Shao L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. ICCV, 2021.

23 Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows[J]. arXiv preprint arXiv:2103.14030, 2021.

24 Zhu X, Su W, Lu L, Li B, Wang X, Dai J. Deformable detr: Deformable transformers for end-to-end object detection[J].

ICLR, 2020.

25 Xie E, Wang W, Yu Z, Anandkumar A, Alvarez J, Luo P. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers[J]. arXiv preprint arXiv:2105.15203, 2021.

26 Cheng B, Misra I, Schwing A G, Kirillov A, Girdhar R. Masked-attention Mask Transformer for Universal Image Segmentation[J]. arXiv preprint arXiv:2112.01527, 2021.

27 Zhou L, Zhou Y, Corso J J, Socher R, Xiong C. End-to-end dense video captioning with masked transformer[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8739-8748.

28 Zeng Y, Fu J, Chao H. Learning joint spatial-temporal transformations for video inpainting[C]//European Conference on Computer Vision. Springer, Cham, 2020: 528-543.

29 Jiang Y, Chang S, Wang Z. Transgan: Two transformers can make one strong gan[J]. arXiv preprint arXiv:2102.07074, 2021, 1(3).

30 Zhao H, Jiang L, Jia J, Torr P H, Koltun V. Point transformer[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 16259-16268.

31 Guo M H, Cai J X, Liu Z N, Mu T-J, Martin R, Hu S M. Pct: Point cloud transformer[J]. Computational Visual Media, 2021, 7(2): 187-199.

32 Shen Z, Zhang M, Zhao H, Yi S, Li H. Efficient attention: Attention with linear complexities[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021: 3531-3539.

33 Katharopoulos A, Vyas A, Pappas N, François F. Transformers are rnns: Fast autoregressive transformers with linear attention[C]//International Conference on Machine Learning. PMLR, 2020: 5156-5165.

34 Guo J, Han K, Wu H, Xu C, Tang Y, Xu C, Wang Y. Cmt: Convolutional neural networks meet vision transformers[J]. arXiv preprint arXiv:2107.06263, 2021.

35 Xiao T, Dollar P, Singh M, Mintun E, Darrell T, Dollár P Girshick R. Early convolutions help transformers see better[J]. Advances in Neural Information Processing Systems, 2021, 34.

36 Kolesnikov A, Beyer L, Zhai X, Puigcerver J, Yung J, Gelly S, Houlsby N. Big transfer (bit): General visual representation learning[C]//Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23 – 28, 2020, Proceedings, Part V 16. Springer International Publishing, 2020: 491-507.

37 Mahajan D, Girshick R, Ramanathan V, He K, Paluri M, Li Y, Bharambe A, Van Der Maaten L. Exploring the limits of weakly supervised pretraining[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 181-196.

38 Touvron H, Vedaldi A, Douze M, Hervé Jégou. Fixing the train-test resolution discrepancy[J]. Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, 2019.

39 Xie Q, Luong M T, Hovy E, Le Q. Self-training with noisy student improves imagenet classification[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 10687-10698.

40 Ba J L, Kiros J R, Hinton G E. Layer normalization[J]. arXiv preprint arXiv:1607.06450, 2016.

41 Kim Y, Denton C, Hoang L, Rush A. Structured attention networks[J]. ICLR, 2017.

42 Buades A, Coll B, Morel J M. A non-local algorithm for image denoising[C]//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). IEEE, 2005, 2: 60-65.

43 Wang X, Girshick R, Gupta A, He K. Non-local neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7794-7803.

44 Han Q, Fan Z, Dai Q, Sun L Cheng M, Liu J, Wang J. Demystifying Local Vision Transformer: Sparse Connectivity, Weight Sharing, and Dynamic Weight[J]. arXiv preprint arXiv:2106.04263, 2021.

45 Chen L C, Papandreou G, Kokkinos I, Murphy K, Yuille A. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(4): 834-848.

46 Jia X, De Brabandere B, Tuytelaars T, Gool L. Dynamic filter networks[J]. Advances in neural information processing systems, 2016, 29: 667-675. arXiv preprint arXiv:2109.03814, 2021.

47 Islam M A, Jia S, Bruce N D B. How much position information do convolutional neural networks encode?[J]. ICLR, 2020.

48 Tay Y, Dehghani M, Bahri D, Metzler D. Efficient transformers: A survey[J]. arXiv preprint arXiv:2009.06732, 2020.

49 Child R, Gray S, Radford A, Sutskever I. Generating long sequences with sparse transformers[J]. arXiv preprint arXiv:1904.10509, 2019.

50 Kitaev N, Kaiser ?, Levskaya A. Reformer: The efficient transformer[J]. ICLR, 2020.

51 Rao Y, Zhao W, Liu B, Lu J, Zhou J, Hsieh C-J. DynamicViT: Efficient Vision Transformers with Dynamic Token Sparsification[J]. arXiv preprint arXiv:2106.02034, 2021.

52 Wang W, Yao L, Chen L, Cai D, He X, Liu W. Crossformer: A versatile vision transformer based on cross-scale attention[J]. arXiv e-prints, 2021: arXiv: 2108.00154.

53 Zhang Q, Yang B. ResT: An Efficient Transformer for Visual Recognition[J]. arXiv preprint arXiv:2105.13677, 2021.

54 Choromanski K, Likhosherstov V, Dohan D, Song X, Gane A, Sarlos T, Hawkins P, Davis J, Mohiuddin A, Kaiser L. Rethinking attention with performers[J]. arXiv preprint arXiv:2009.14794, 2020.

55 Tsai Y H H, Bai S, Yamada M, Morency L-P Salakhutdinov R. Transformer Dissection: A Unified Understanding of Transformer's Attention via the Lens of Kernel[J]. arXiv preprint arXiv:1908.11775, 2019.

56 Zhai S, Talbott W, Srivastava N, Huang C, Goh H, Zhang R, Susskind J. An Attention Free Transformer[J]. arXiv preprint, 2021.

57 Lu J, Yao J, Zhang J, Zhu X, Xu H, Gao W, Xu C, Xiang T Zhang L. Soft: Softmax-free transformer with linear complexity[J]. Advances in Neural Information Processing Systems, 2021, 34.

58 Touvron H, Cord M, Douze M, Francisco M, Alexandre S, Hervé Jégou. Training data-efficient image transformers & distillation through attention[C]//International Conference on Machine Learning. PMLR, 2021: 10347-10357.

59 Zhou D, Shi Y, Kang B, Yu W, Jiang Z, Li Y, Jin X, Hou Q, Feng J. Refiner: Refining Self-attention for Vision Transformers[J]. arXiv preprint arXiv:2106.03714, 2021.

60 d'Ascoli S, Touvron H, Leavitt M, Morcos A, Biroli G, Sagun L. Convit: Improving vision transformers with soft convolutional

inductive biases[J]. ICML.10697, 2021.

61 Li Y, Zhang K, Cao J, Timofte R, Van Gool L. Localvit: Bringing locality to vision transformers[J]. arXiv preprint arXiv:2104.05707, 2021.

62 Lin T Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

63 Chen C F, Fan Q, Panda R. Crossvit: Cross-attention multi-scale vision transformer for image classification[J]. arXiv preprint arXiv:2103.14899, 2021.

64 Gong C, Wang D, Li M, Chandra V, Liu Q. Improve vision transformers training by suppressing over-smoothing[J]. arXiv preprint arXiv:2104.12753, 2021.

65 Yun S, Han D, Oh S J, Chun S, Choe J Yoo Y. Cutmix: Regularization strategy to train strong classifiers with localizable features[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 6023-6032.

66 Zhou D, Kang B, Jin X, Yang L, Lian X, Jiang Z, Hou Q, Feng J. Deepvit: Towards deeper vision transformer[J]. arXiv preprint arXiv:2103.11886, 2021.

67 Tay Y, Bahri D, Metzler D, Juan D-C, Zhao Z,Zheng C. Synthesizer: Rethinking self-attention for transformer models[C]//International Conference on Machine Learning. PMLR, 2021: 10183-10192.

68 Yuan L, Hou Q, Jiang Z, Feng J, Yan S. Volo: Vision outlooker for visual recognition[J]. arXiv preprint arXiv:2106.13112, 2021.

69 Mihcak M K, Kozintsev I, Ramchandran K, Moulin P. Low-complexity image denoising based on statistical modeling of wavelet coefficients[J]. IEEE Signal Processing Letters, 1999, 6(12): 300-303.

70 He K, Sun J, Tang X. Guided image filtering[J]. IEEE transactions on pattern analysis and machine intelligence, 2012, 35(6): 1397-1409.

71 Criminisi A, Pérez P, Toyama K. Region filling and object removal by exemplar-based image inpainting[J]. IEEE Transactions on image processing, 2004, 13(9): 1200-1212.

72 Raghu M, Unterthiner T, Kornblith S, Zhang C, Dosovitskiy A. Do Vision Transformers See Like Convolutional Neural Networks?[J]. Advances in Neural Information Processing Systems, 2021, 34.

73 Yuan K, Guo S, Liu Z, Zhou A, Yu F, Wu W. Incorporating convolution designs into visual transformers[J]. arXiv preprint, 2021.

74 Chen Y, Dai X, Chen D, Liu M, Dong X, Yuan L, Liu Z. Mobile-former: Bridging mobilenet and transformer[J]. arXiv preprint arXiv:2108.05895, 2021.

75 Mehta S, Rastegari M. MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer[J]. arXiv preprint arXiv:2110.02178, 2021.

76 Peng Z, Huang W, Gu S, Xie L, Wang Y, Jiao J, Ye Q. Conformer: Local Features Coupling Global Representations for Visual Recognition[J]. arXiv preprint arXiv:2105.03889, 2021.

77 Yan H, Deng B, Li X, Qiu X. TENER: adapting transformer encoder for named entity recognition[J]. arXiv preprint arXiv:1911.04474, 2019.

78 Shaw P, Uszkoreit J, Vaswani A. Self-attention with relative position representations[J]. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2018.

79 Dai Z, Yang Z, Yang Y, Yang Y, Carbonell J, Le Q V, Salakhutdinov R. Transformer-xl: Attentive language models beyond a fixed-length context[J]. arXiv preprint arXiv:1901.02860, 2019.

80 Huang Z, Liang D, Xu P, Xiang B. Improve transformer models with better relative position embeddings[J]. Findings of the Association for Computational Linguistics: EMNLP, 2020.

81 Ramachandran P, Parmar N, Vaswani A, Bello I, Levskaya A, Shlens J. Stand-alone self-attention in vision models[J]. Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, 2019.

82 Wu K, Peng H, Chen M, Fu J, Chao H. Rethinking and improving relative position encoding for vision transformer[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 10033-10041.

83 Deng J, Dong W, Socher R, LI L, Li K, Li F. Imagenet: A large-scale hierarchical image database[C]//2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009: 248-255.

84 Zhao Y, Wang G, Tang C, Luo C, Zeng W, Zha Z. A Battle of Network Structures: An Empirical Study of CNN, Transformer, and MLP [J]. arXiv preprint arXiv:2108.13002, 2021.

85 Tolstikhin I O, Houlsby N, Kolesnikov A, Beyer L,Zhai X,Unterthiner T,Yung J,Steiner A,Keysers D,Uszkoreit J,Lucic M,Dosovitskiy A. Mlp-mixer: An all-mlp architecture for vision[J]. Advances in Neural Information Processing Systems, 2021, 34.

86 Touvron H, Bojanowski P, Caron M, Caron M,Cord M,El-Nouby A,Grave E,Izacard G,Joulin A,Synnaeve G,Verbeek J,Jégou H. Resmlp: Feedforward networks for image classification with data-efficient training[J]. arXiv preprint arXiv:2105.03404, 2021.

87 Liu H, Dai Z, So D,V.Le Q. Pay attention to MLPs[J]. Advances in Neural Information Processing Systems, 2021, 34.

88 Chen S, Xie E, Ge C,Chen R,Liang D,Luo P. Cyclemlp: A mlp-like architecture for dense prediction[J]. arXiv preprint arXiv:2107.10224, 2021.

89 Ng D, Chen Y, Tian B, Fu Q,Chng ES. ConvMixer: Feature Interactive Convolution with Curriculum Learning for Small Footprint and Noisy Far-field Keyword Spotting[J]. arXiv preprint arXiv:2201.05863, 2022.

90 Yu W, Luo M, Zhou P, Si C, Zhou Y, Wang X, Feng J, Yan S. Metaformer is actually what you need for vision[J]. arXiv preprint arXiv:2111.11418, 2021.

91 Jeevan P,Sethi A. Convolutional Xformers for Vision[J]. arXiv preprint arXiv:2201.10271, 2022.

92 Liu Z, Mao H, Wu Chao-Yuan, Feichtenhofer C, Darrell T, Xie S. A ConvNet for the 2020s[J]. arXiv preprint arXiv:2201.03545, 2022.

93 Ding X, Zhang X, Zhou Y, Han J, Ding G, Sun J. Scaling Up Your Kernels to 31x31: Revisiting Large Kernel Design in CNNs[J]. arXiv preprint arXiv:2203.06717, 2022.

94 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. ICLR, 2014.

95 Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 28: 91-99.

96 Chen H, Sun K, Tian Z, Shen C, Huang Y, Yan Y. BlendMask: Top-down meets bottom-up for instance segmentation[C]//Proceedings of the IEEE/CVF conference on com-

puter vision and pattern recognition. 2020: 8573-8581.

97 Wang X, Zhang R, Kong T, Li L, Shen C. SOLOv2: Dynamic and fast instance segmentation[J]. Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, 2020.

98 Dong B, Zeng F, Wang T, Zhang X, Wei Y. SOLQ: Segmenting Objects by Learning Queries[J]. arXiv preprint arXiv:2106.02351, 2021.

99 He K, Gkioxari G, Dollár P, Girshick R B. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.

100 Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg A C. Ssd: Single shot multibox detector[C]//European conference on computer vision. Springer, Cham, 2016: 21-37.

101 Law H, Deng J. Cornernet: Detecting objects as paired keypoints[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 734-750.

102 Zhou X, Wang D, Krähenbühl P. Objects as points[J]. arXiv preprint arXiv:1904.07850, 2019.

103 Lin T Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár, P, Zitnick, C L. Microsoft coco: Common objects in context[C]//European conference on computer vision. Springer, Cham, 2014: 740-755.

104 Lin T Y, Goyal P, Girshick R, He K, Doll P. Focal loss for dense object detection[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.

105 Cai Z, Vasconcelos N. Cascade r-cnn: Delving into high quality object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6154-6162.

106 Zhang S, Chi C, Yao Y, Lei Z, Li S. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 9759-9768.

107 Chen Y, Zhang Z, Cao Y, Wang L, Lin S, Hu H. Reppoints v2: Verification meets regression for object detection[J]. Advances in Neural Information Processing Systems, 2020, 33.

108 Sun P, Zhang R, Jiang Y, Kong T, Xu, C, Zhan W, Tomizuka M, Li L, Yuan Z, Wang C. Sparse r-cnn: End-to-end object detection with learnable proposals[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 14454-14463.

109 Yao Z, Ai J, Li B, Zhang C. Efficient DETR: Improving End-to-End Object Detector with Dense Prior[J]. arXiv preprint arXiv:2104.01318, 2021.

110 Gao P, Zheng M, Wang X, Dai J, Li H. Fast convergence of detr with spatially modulated co-attention[J]. arXiv preprint arXiv:2101.07448, 2021.

111 Meng D, Chen X, Fan Z, Zeng G, Li H, Yuan Y, Sun L, Wang J. Conditional detr for fast training convergence[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 3651-3660.

112 Dai Z, Cai B, Lin Y, Chen J. Up-detr: Unsupervised pre-training for object detection with transformers[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 1601-1610.

113 Zheng M, Gao P, Zhang R, Li K, Wang X, Li H, Dong H. End-to-end object detection with adaptive clustering transformer[J]. arXiv preprint arXiv:2011.09315, 2020.

114 Fang Y, Liao B, Wang X, Fang J, Qi J, Wu R, Niu J, Liu W. You Only Look at One Sequence: Rethinking Transformer in Vision through Object Detection[J]. arXiv preprint arXiv:2106.00666, 2021.

115 Sun Z, Cao S, Yang Y, Kitani K. Rethinking transformer-based set prediction for object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 3611-3620.

116 Tian Z, Shen C, Chen H, He T. Fcos: Fully convolutional one-stage object detection[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 9627-9636.

117 Beal J, Kim E, Tzeng E, Park, Dong H, Zhai A, Kislyuk D. Toward transformer-based object detection[J]. arXiv preprint arXiv:2012.09958, 2020.

118 Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.

119 Minaee S, Boykov Y Y, Porikli F, Plaza A, Kehtarnavaz N, Terzopoulos D. Image segmentation using deep learning: A survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021.

120 Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.

121 Zheng S, Lu J, Zhao H, Zhu X, Luo Z, Wang Y, Fu Y, Feng J, Xiang T, Torr P. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 6881-6890.

122 Xiao T, Liu Y, Zhou B, Jiang Y, Sun J. Unified perceptual parsing for scene understanding[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 418-434.

123 Wang H, Zhu Y, Adam H, Yuille A, Chen L. Max-deeplab: End-to-end panoptic segmentation with mask transformers[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 5463-5474.

124 Strudel R, Garcia R, Laptev I, Schmid C. Segmenter: Transformer for Semantic Segmentation[J]. arXiv preprint arXiv:2105.05633, 2021.

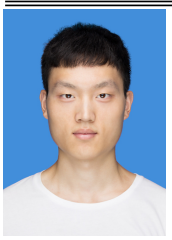
125 Cheng B, Schwing A, Kirillov A. Per-pixel classification is not all you need for semantic segmentation[J]. Advances in Neural Information Processing Systems, 2021, 34.

126 Fang Y, Yang S, Wang X, Li Y, Fang C, Shan Y, Feng B, Liu W. QueryInst: Parallelly Supervised Mask Query for Instance Segmentation[J]. arXiv preprint arXiv:2105.01928, 2021.

127 Hu J, Cao L, Lu Y, Zhang S C Wang Y, Li K, Huang F, Shao L, Ji R. ISTR: End-to-End Instance Segmentation with Transformers[J]. arXiv preprint arXiv:2105.00637, 2021.

128 Li Z, Wang W, Xie E, Yu Z, Anandkumar A, Alvarez J, Lu T, Luo P. Panoptic SegFormer[J].

129 Wang H, Zhu Y, Green B, Adam H, Yuille A, Chen L C. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation[C]//European Conference on Computer Vision. Springer, Cham, 2020: 108-126.



田永林 中国科学技术大学与中科院自动化研究所联合培养博士研究生. 2017 年获得中国科学技术大学自动化系学士学位. 主要研究方向为计算机视觉, 智能交通.
E-mail: tyldyx@mail.ustc.edu.cn (**TIAN Yong-Lin** Ph. D. candidate in Department of Automation, University of Science and Technology of China and Institute of Automation, Chinese Academy of Sciences. He received his bachelor degree from University of Science and Technology of China in 2017. His research interest covers computer vision and intelligent transportation system.)

王雨桐 中国科学院自动化研究所复杂系统管理与控制国家重点实验室助理研究员. 2016 年获得中国科学院大学控制理论与控制工程专业博士学位. 主要研究方向为深度学习中的对抗攻击与防御.
E-mail: yutong.wang@ia.ac.cn (**WANG Yu-Tong** Assistant professor at the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. She received her Ph. D. degree in control theory and control engineering from University of Chinese Academy of Sciences in 2021. Her research interest covers computer vision and adversarial attack.)

王建功 中国科学院自动化研究所博士研究生. 2018 年获得同济大学学士学位. 主要研究方向为计算机视觉, 交通场景理解, 医学图像处理.
E-mail: wangjiangong2018@ia.ac.cn (**WANG Jian-Gong** Ph. D. candidate in Institute of Automation, Chinese Academy of Sciences. He received his bachelor degree from Tongji University in 2018. His research interest covers computer vision, traffic scene understanding and medical image processing.)

王晓 中国科学院自动化研究所复杂系统管理与控制国家重点实验室副研究员, 青岛智能产业技术研究院院长. 2016 年获得中国科学院大学社会计算博士学位。主要研究方向为社会交通, 动态网群组织, 平行智能和社交网络分析.
E-mail: x.wang@ia.ac.cn (**WANG Xiao** Associate professor at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences and the executive president of the Qingdao Academy of Intelligent Industries. Her research interest covers social transportation, cybermovement organizations, parallel intelligence and social network analysis.)

王飞跃 中国科学院自动化研究所复杂系统管理与控制国家重点实验室研究员. 主要研究方向为智能系统和复杂系统的建模、分析与控制. 本文通信作者.
E-mail: feiyue.wang@ia.ac.cn (**WANG Fei-Yue** Professor at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interest covers modeling, analysis, and control of intelligent systems and complex systems. Corresponding author of this paper.)