视觉中的 Transformer：一篇综述

Transformers in Vision: A Survey

Salman Khan、Muzammal Naseer、Munawar Hayat、Syed Waqas Zamir、Fahad Shahbaz Khan 和 Mubarak Shah

arXiv:2101.01169v5 [cs.CV]，2022 年 1 月 19 日

摘要

Transformer 模型在自然语言任务上取得了惊人的结果，促使视觉研究者探索它们在计算机视觉问题中的应用。Transformer 的突出优点之一是能够建模输入序列各元素之间的长程依赖；与长短期记忆网络（LSTM）等循环网络相比，它还支持并行处理序列。不同于卷积网络，Transformer 的设计只需要很少的归纳偏置，并且天然适合作为集合函数。此外，Transformer 设计简明，可以用相似的处理模块处理图像、视频、文本和语音等多种模态，而且可很好地扩展到超大容量网络和海量数据集。这些优势使基于 Transformer 的诸多视觉任务取得了令人振奋的进展。本文旨在全面梳理计算机视觉领域的 Transformer 模型。我们首先介绍支撑 Transformer 成功的基本概念，即自注意力、大规模预训练和双向特征编码。随后考察 Transformer 在视觉中的广泛应用，包括常见的识别任务（如图像分类、目标检测、动作识别和分割）、生成建模、多模态任务（如视觉问答、视觉推理和视觉指代定位）、视频处理（如活动识别和视频预测）、低层视觉（如图像超分辨率、图像增强和上色）以及三维分析（如点云分类和分割）。我们从架构设计和实验表现两方面比较常用技术各自的优点和局限。最后，我们分析尚未解决的研究问题及可能的后续工作，希望借此激发更多研究者的兴趣，共同解决 Transformer 在计算机视觉应用中面临的挑战。

关键词 自注意力、Transformer、双向编码器、深度神经网络、卷积网络、自监督。

作者单位与稿件信息

S. Khan、M. Naseer 和 F. S. Khan 就职于阿联酋阿布扎比的穆罕默德 • 本 • 扎耶德人工智能大学。电子邮件：firstname.lastname@mbzuai.ac.ae。

M. Hayat 就职于澳大利亚莫纳什大学信息技术学院，地址为 Clayton VIC 3800。

S. W. Zamir 就职于阿联酋阿布扎比的 Inception 人工智能研究院。

S. Khan 和 M. Naseer 也就职于澳大利亚国立大学工程与计算机科学学院（CECS），地址为 Canberra ACT 0200。

F. S. Khan 也就职于瑞典林雪平大学计算机视觉实验室。

M. Shah 就职于美国中佛罗里达大学计算机视觉研究中心，地址为 Orlando, FL 32816。

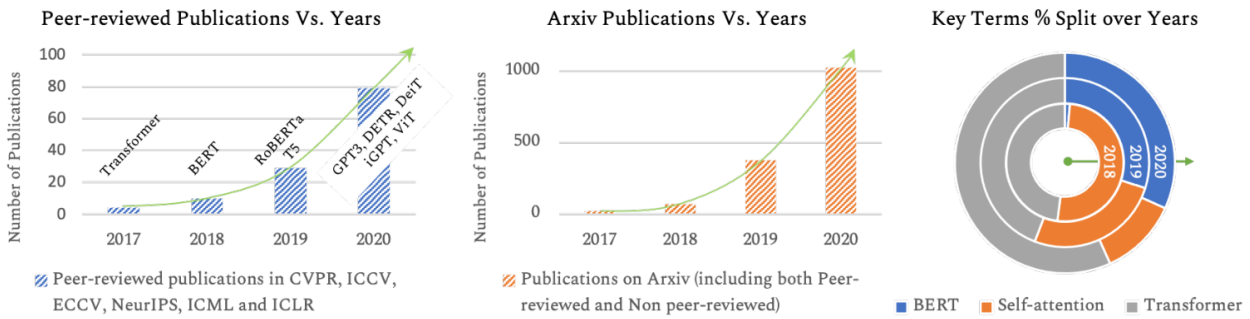
稿件于 2021 年 3 月收到。

1 引言

近期，Transformer 模型 [1] 在文本分类、机器翻译 [2] 和问答等多种语言任务上表现出色。其中广为采用的包括 BERT（来自 Transformer 的双向编码器表征）[3]、GPT（生成式预训练 Transformer）v1-3 [4]–[6]、RoBERTa（稳健优化的 BERT 预训练）[7] 和 T5（文本到文本迁移 Transformer）[8]。随着 Transformer 可以扩展到极大容量，其深远影响愈发明显 [9], [10]。例如，拥有 1750 亿个参数（175 billion）的 GPT-3 [6] 显著超越了拥有 3.4 亿个参数（340 million）的 BERT-large [3]；近期的混合专家模型 Switch Transformer [10] 更是扩展到惊人的 1.6 万亿个参数（1.6 trillion）！

Transformer 网络在自然语言处理（NLP）领域取得的突破，引起了计算机视觉研究者的浓厚兴趣，促使他们将这类模型用于视觉和多模态学习任务（图 1）。

图 1 关键词发文趋势 · 原文 p.2



统计过去几年中，计算机视觉与机器学习领域同行评议论文及 arXiv 论文标题中 BERT、自注意力和 Transformer 等关键词的出现次数。图中的曲线显示近期相关文献持续增长。本文梳理 Transformer 在计算机视觉领域的近期进展。

图中术语 | Peer-reviewed publications＝同行评议论文；Number of publications＝论文数量；Key terms % split＝关键词占比。左、中两图计数，右图显示词项比例。

不过，视觉数据具有特定结构（例如空间与时间上的连贯性），因而需要新的网络设计和训练方案。由此产生的 Transformer 模型及其变体已成功用于图像识别 [11], [12]、目标检测 [13], [14]、分割 [15]、图像超分辨率 [16]、视频理解 [17], [18]、图像生成 [19]、文本到图像合成 [20]、视觉问答 [21], [22] 等诸多场景 [23]–[26]。本文旨在梳理计算机视觉领域近期这些令人振奋的工作，为感兴趣的读者提供全面的参考。

Transformer 架构以自注意力机制为基础，用以学习同一序列元素之间的关系。循环网络递归地处理序列元素，只能关注短程上下文；Transformer 则能关注完整序列，由此学习长程关系。尽管注意力模型已广泛用于前馈和循环网络 [27], [28]，Transformer 完全基于注意力机制，并采用了一种专门针对并行运算优化的实现，即多头注意力。这些模型的一个重要特性是能扩展到高复杂度模型和大规模数据集；相比之下，硬注意力 [29] 等其他方案具有随机性，需要用蒙特卡洛方法采样注意位置。相较于卷积网络和循环网络 [30]–[32]，Transformer 对问题结构只作很少的先验假设，因此通常在大规模（无标签）数据集上借助代理任务预训练 [1], [3]。这种做法避免了昂贵的人工标注，并能编码表达能力强、可泛化的表征，把数据集中各实体间丰富的关系纳入模型。随后在下游任务中对所得表征进行有监督微调，以取得良好的结果。

本文全面介绍为计算机视觉任务开发的 Transformer 模型。我们对网络设计空间进行分类，并指出现有方法的主要优势和不足。既有综述主要关注 NLP [33], [34]，或讨论一般的注意力方法 [27], [33]。本文聚焦新兴的视觉 Transformer 领域，按照自注意力的内在特征和所研究的任务，对近期方法作系统梳理。我们先介绍 Transformer 网络的核心概念，再详述近期视觉 Transformer 的具体设计。只要适合比较，我们就将 NLP 所用的 Transformer [1] 与针对视觉问题开发的 Transformer 对照，以突显主要创新和有趣的领域洞见。近期工作表明，卷积运算既可完全由基于注意力的 Transformer 模块替代，也可与后者用在同一架构中，发挥两类互补运算的协同作用。最后，本文列出现有的开放问题，并展望可能的后续工作。

2 基础

传统 Transformer 模型的发展受益于两个关键思想。(a) 首先是自注意力。与难以编码长程关系的传统循环模型相比，自注意力能捕获序列元素之间的“长程”依赖。(b) 第二个是先在大规模（有或无标签）语料上以（自）监督方式预训练[1]，再使用小规模有标签数据集针对目标任务微调 [3], [7], [38]。下文简要讲解这两个思想（第 2.2 节和第 2.1 节），同时概述应用了这些思想的开创性 Transformer 网络（第 2.3 和 2.4 节）。这些背景有助于理解后续用于计算机视觉的 Transformer 模型（第 3 节）。

原文脚注 1. 近期已有几种视觉 Transformer 表明，模型在 ImageNet-1K 上无须专门的预训练阶段，也可以端到端训练 [35]–[37]。不过，其性能一般仍低于经过预训练的对应模型。

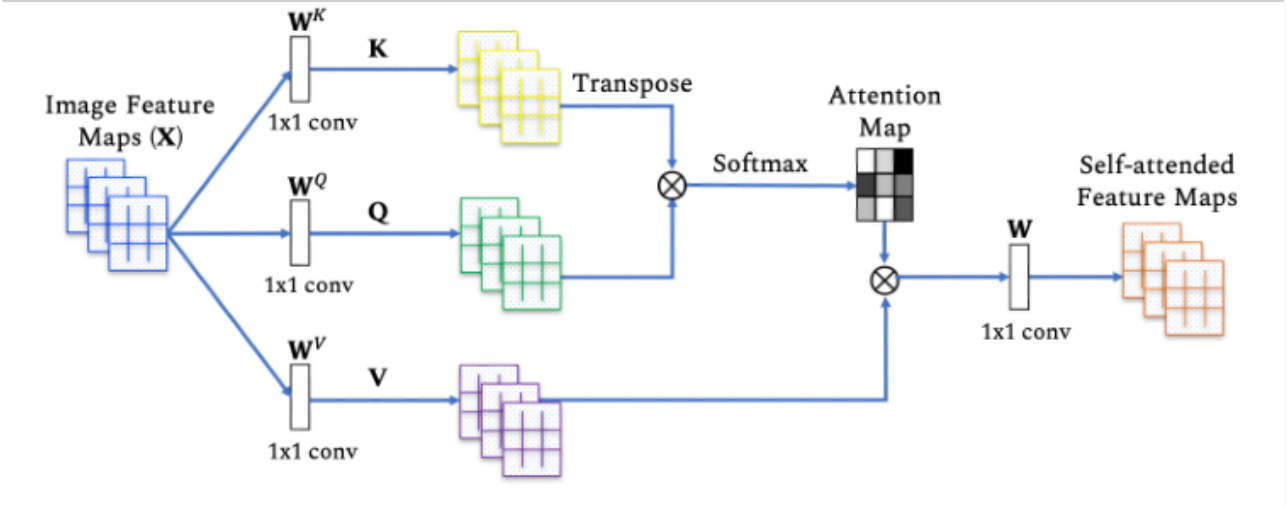
2.1 Transformer 中的自注意力

给定一个序列，自注意力估计其中某一元素与其他元素的相关性，例如句子中哪些单词可能一起出现。自注意力是 Transformer 的核心组件；在结构化预测任务中，它显式建模序列中所有实体之间的相互作用。简而言之，一个自注意力层把整个输入序列的全局信息汇聚起来，以更新序列的各个部分。以 $\mathbf{X} \in \mathbb{R}^{n \times d}$ 表示由 n 个实体 $(x_1, x_2, \cdots, x_n)$ 组成的序列，其中 d 是表示每个实体的嵌入维度。自注意力以全局上下文信息为依据编码每个实体，从而捕获所有 n 个实体之间的相互作用。为此定义三个可学习的权重矩阵，分别变换查询（$\mathbf{W}^Q \in \mathbb{R}^{d \times d_q}$）、键（$\mathbf{W}^K \in \mathbb{R}^{d \times d_k}$）和值（$\mathbf{W}^V \in \mathbb{R}^{d \times d_v}$），其中 $d_q = d_k$。先将输入序列 $\mathbf{X}$ 投影到这些权重矩阵上，得到 $\mathbf{Q} = \mathbf{X}\mathbf{W}^Q$、$\mathbf{K} = \mathbf{X}\mathbf{W}^K$ 和 $\mathbf{V} = \mathbf{X}\mathbf{W}^V$。自注意力层的输出 $\mathbf{Z} \in \mathbb{R}^{n \times d_v}$ 如下：

$$\mathbf{Z} = \mathbf{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_q}}\right)\mathbf{V}.$$

对于序列中给定的实体，自注意力先计算其查询与所有键的点积，再用 softmax 运算归一化得到注意力分数。此后各实体表示变为序列中所有实体的加权和，权重即上述注意力分数（图 2 和图 3 最上排的左侧模块）。

图 2 视觉中的单头自注意力 · 原文 p.2



视觉领域使用的自注意力模块示例 [39]。给定输入的图像特征序列，先计算（键、查询、值）三元组，再计算注意力并用其重新加权值。图中只显示一个注意力头，最后通过输出投影 (W) 获得与输入维度相同的输出特征。图片改绘自 [39]。

图中术语 | Image Feature Maps＝图像特征图；Attention Map＝注意力图；Transpose＝转置；Self-attended Feature Maps＝自注意力处理后的特征图。1×1 conv 表示逐点卷积。

带掩码的自注意力：标准自注意力层关注所有实体。Transformer 模型 [1] 的训练目标是预测序列中的下一个实体，其解码器中使用的自注意力模块带有掩码，因而不能关注序列中后续的实体。具体做法是与掩码 $\mathbf{M} \in \mathbb{R}^{n \times n}$ 作逐元素相乘，其中 M 为上三角矩阵。带掩码的自注意力定义为：

$$\textbf{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_q}} \circ \mathbf{M}\right),$$

其中 ∘ 表示 Hadamard 积。简而言之，在预测序列中的某个实体时，带掩码的自注意力将后续实体的注意力分数设为零。

多头注意力：为囊括序列元素之间多种复杂的关系，多头注意力包含若干自注意力模块（原始 Transformer 模型 [1] 中 h = 8）。每个模块都拥有自己的一组可学习权重矩阵 $\{W_i^Q, W_i^K, W_i^V\}$，其中 i = 0 ⋯ (h−1)。给定输入 X，将多头注意力中 h 个自注意力模块的输出拼接成单个矩阵 $[Z_0, Z_1, \cdots, Z_{h-1}] \in \mathbb{R}^{n \times h \cdot d_v}$，再用权重矩阵 $W \in \mathbb{R}^{h \cdot d_v \times d}$ 进行投影（图 3 最上排）。

自注意力与卷积的主要区别在于，前者的滤波器是动态计算出来的，而卷积所用的是静态滤波器（对任意输入都保持不变）。此外，自注意力对排列变化和输入点数量的变化具有不变性。因此，它能轻松处理不规则输入，而标准卷积要求输入具有网格结构。文献还从理论上表明，自注意力（带位置编码）是一种更灵活的运算，可以模拟卷积模型编码局部特征的行为 [40]。Cordonnier 等人 [41] 进一步研究了自注意力和卷积的关系。他们的实验结果证实，（具有足够参数的）多头自注意力是更通用的运算，可以把卷积的表达能力囊括为一种特例。实际上，自注意力既能学习全局特征，又能学习局部特征，而且能够自适应地学习核权重和感受野，类似于可变形卷积 [42]。

2.2 （自）监督预训练

基于自注意力的 Transformer 模型通常采用两阶段训练机制。首先，在大规模数据集（有时是几个现成数据集的组合 [22], [43]）上进行有监督 [11] 或自监督预训练 [3], [44], [45]。随后，用中小规模数据集调整预训练的权重，以适配下游任务。这些任务包括图像分类 [46]、目标检测 [13]、零样本分类 [20]、问答 [10] 和动作识别 [18]。语言和视觉领域均有工作指出预训练对大规模 Transformer 的有效性。例如，视觉 Transformer（ViT-L）[11] 如果仅在 ImageNet 训练集上训练，与先在含 3 亿张图像（300 million）的 JFT [47] 数据集上预训练相比，在 ImageNet 测试集上的准确率会下降 13 个百分点。

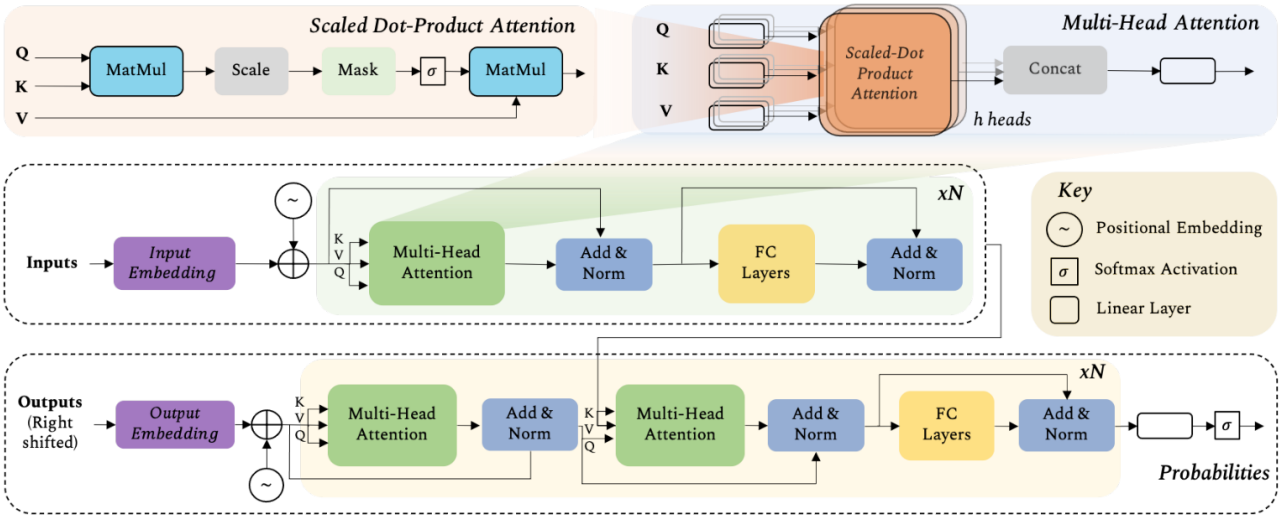
由于大规模获取人工标签十分繁琐，自监督学习已在预训练阶段得到有效利用。自监督预训练对于充分发挥 Transformer 网络的可扩展性和泛化性起到了关键作用，使训练参数超过万亿的网络也成为可能（例如 Google 近期的 Switch Transformer [10]）。文献 [48], [49] 全面综述了自监督学习（SSL）。Y. LeCun [50] 将其基本思想概括为“填空”，即尝试预测图像中被遮挡的数据、视频时序中的未来帧或过去帧，或完成代理任务，例如预测输入的旋转幅度、图像块的排列方式或灰度图像的颜色。对比学习是施加自监督约束的另一种有效手段。在这里，对同一图像施加变换，得到两类修改版本：一类不改变其类别语义（例如图像风格化、裁剪）；另一类产生语义变化（例如把同一场景中的目标换成另一个目标，或对图像施加微小的对抗扰动以改变其类别）。随后训练模型，使其不受无关变换影响，同时着重建模那些可能改变语义标签的细微变化。

自监督学习是一种很有前景的范式，因为它能利用大量易得的无标注数据进行学习。在基于 SSL 的预训练阶段，模型通过解决代理任务来学习数据中有意义的表征。代理任务的伪标签根据数据属性和任务定义自动生成，无须昂贵的人工标注。因此，怎样定义代理任务是 SSL 中的一项关键决策。按代理任务划分，现有 SSL 方法大致有三类：(a) 给定条件输入后合成图像或视频的生成式方法；(b) 利用图像块或视频帧之间关系的上下文方法；(c) 利用多种数据模态的跨模态方法。生成式方法的例子有掩码图像建模 [43]、图像上色 [51] 等条件生成任务，也包括图像超分辨率 [52]、图像修补 [53] 及基于 GAN 的方法 [54], [55]。上下文代理任务包括完成由图像块构成的拼图 [56]–[58]、对被掩蔽的目标做分类 [22]、预测旋转等几何变换 [46], [59] 或验证视频帧的时间顺序 [60]–[62]。跨模态代理任务验证两种输入模态是否对应，例如文本与图像 [63]、音频与视频 [64], [65]，或 RGB 与光流 [66]。

2.3 Transformer 模型

图 3 展示了文献 [1] 提出的 Transformer 模型架构。它采用编码器–解码器结构。编码器（中间一排）由六个相同的模块组成（图 3 中 N = 6），每个模块有两个子层：一个多头自注意力网络，以及一个简单的逐位置全连接前馈网络。如图 3 所示，每个模块之后都使用残差连接 [67] 和层归一化 [68]。常规卷积网络同时进行特征汇聚和特征变换（例如卷积层后接非线性运算）；而 Transformer 将两步解耦，自注意力层只做汇聚，前馈层做变换。与编码器类似，解码器（最下排）也由六个相同的模块组成。各解码器模块含三个子层，其中两个（多头自注意力和前馈层）与编码器类似，第三个对相应编码器模块的输出执行多头注意力，如图 3 所示。

图 3 原始 Transformer 的编码器与解码器 · 原文 p.3



Transformer 模型架构 [1]。该模型最初用于语言翻译任务，将一种语言的输入序列转换为另一种语言的输出序列。Transformer 编码器（中间一排）处理源语言序列，先将其转换为嵌入，再交给编码器模块。Transformer 解码器（最下排）利用此前生成的目标语言输出，以及中间支路编码后的输入序列，生成输出序列中的下一个词。作为解码器输入的先前输出序列，通过将目标句右移一位、在句首插入句首词元得到。右移可以防止模型简单地把解码器输入抄到输出。训练所用真值即不作右移、但在末尾加入句末词元的目标语言序列。多头注意力（最上排）与前馈层构成的模块，在编码器和解码器中均重复 N 次。

图中术语｜MatMul＝矩阵乘法；Scale＝缩放；Mask＝掩码；Concat＝拼接；Add & Norm＝残差相加与归一化；FC Layers＝全连接层；Right shifted＝右移。

文献 [1] 的原始 Transformer 模型针对机器翻译任务训练。编码器输入是某一种语言的词序列（句子）。输入序列加入位置编码，以捕捉各词在序列中的相对位置。位置编码和输入具有相同的维度 d = 512，可以学习得到，也可以预先定义，例如用正弦或余弦函数。Transformer [1] 的解码器采用自回归方式，利用此前的预测来输出序列中的下一个词。因此，解码器同时接收编码器的输出和之前的输出，用于预测译文句子中的下一个词。为便于使用残差连接，所有层的输出维度保持一致，即 d = 512。多头注意力中查询、键和值的权重矩阵维度分别设为 $d_q = 64$、$d_k = 64$、$d_v = 64$。

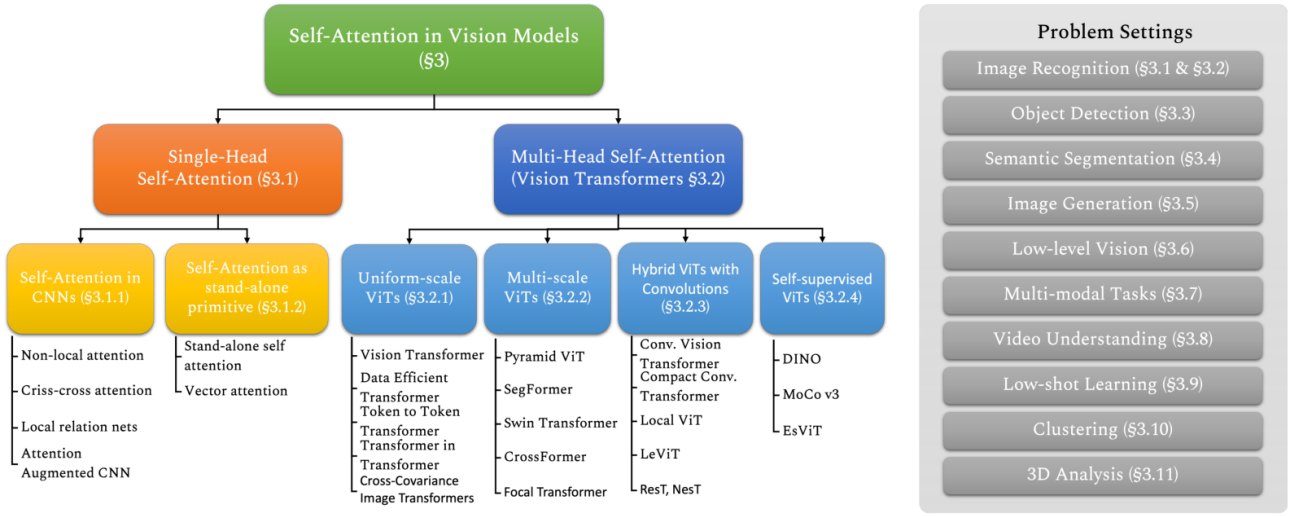
2.4 双向表征

原始 Transformer 模型 [1] 的训练策略，使其只能关注句中某个词左侧的上下文。这是一项局限，因为对大多数语言任务而言，左右两侧的上下文信息都很重要。BERT（来自 Transformer 的双向编码器表征）[3] 提出同时编码句中某个词左右两侧的上下文，从而以自监督方式改善学到的文本数据特征表征。为此，BERT [3] 引入了两种代理任务，对 Transformer 模型 [1] 进行自监督预训练：掩码语言建模和下一句预测。为了将预训练模型适配到下游任务，在预训练模型上加一个面向该任务的额外输出模块，再对整个模型进行端到端微调。下面简要说明这两个代理任务。(1) 掩码语言建模（MLM）：随机掩蔽句中固定比例（15%）的词，用交叉熵损失训练模型预测这些词。模型在预测被掩蔽词的过程中学会利用双向上下文。(2) 下一句预测（NSP）：给出两个句子，模型预测一个二元标签，表示该句对在原文档中是否确实相邻。训练数据可方便地用任意单语语料生成：构造句对 A 和 B，其中 50% 的情况下，B 是原文中 A 的下一句，另 50% 的情况下，B 是随机选取的句子。NSP 使模型能够捕捉句子之间的关系，这对问答和自然语言推断等许多语言建模任务至关重要。

3 视觉中的自注意力与 Transformer

我们将采用自注意力的视觉模型大致分为两类：采用单头自注意力的模型（第 3.1 节），以及在架构中使用基于多头自注意力的 Transformer 模块的模型（第 3.2 节）。下文先讨论基于单头自注意力的第一类框架。这些方法通常在 CNN 架构内使用全局或局部自注意力，或利用矩阵分解提高设计效率，并使用向量化的注意力模型。之后，第 3.2 节讨论基于 Transformer 的视觉架构。

图 4 自注意力设计空间 · 原文 p.5



自注意力的设计空间分类。现有基于自注意力的方法为视觉任务探索单头或多头（Transformer）设计。已有一些值得关注的工作借鉴卷积架构的知识改进 ViT，例如多尺度架构与混合架构。本文后续章节依据自注意力模块的类型（左侧树形图）和计算机视觉的主要任务（右侧）组织。

图中术语 | 左侧为设计分类：单头包含 CNN 内部注意力与独立原语；多头包括等尺度、多尺度、卷积混合、自监督四个分支。右侧为任务分类。

3.1 单头自注意力

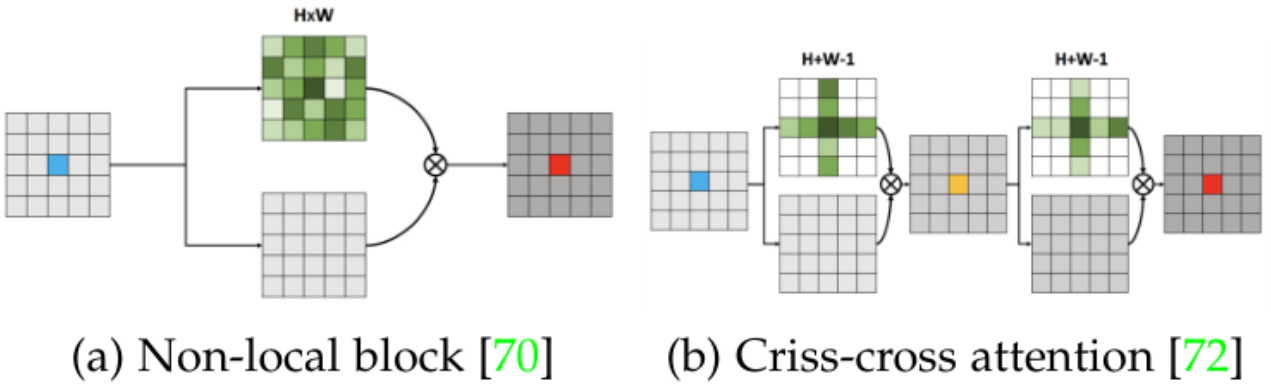
3.1.1 CNN 中的自注意力

受最初主要用于图像去噪的非局部均值运算 [69] 启发，Wang 等人 [70] 为深度神经网络提出一种可微的非局部运算，以前馈方式捕获空间和时间上的长程依赖。给定一张特征图，该算子 [70] 将某一位置的响应计算为特征图上所有位置特征的加权和。这样，非局部运算即可捕获特征图中任意两个位置之间的相互作用，而不受距离限制。视频分类就是一个例子，其中像素在空间和时间上都存在长程相互作用。文献 [70] 证明，凭借长程相互作用建模的能力，非局部深度神经网络能在 Kinetics 数据集 [71] 上实现更准确的视频分类。

自注意力虽然能对全图上下文信息建模，却同时消耗大量显存和算力。如图 5(a) 所示，为给定像素位置编码全局上下文时，非局部模块 [70] 会计算稠密的注意力图（绿色）。非局部模块 [70] 的复杂度高达 $O(N^2)$，其中 N 表示输入特征图的数量。为降低这一计算负担，Huang 等人 [72] 提出十字交叉注意力模块；如图 5(b) 所示，对于每个像素位置，只沿十字交叉路径生成稀疏注意力图。此外，反复应用十字交叉注意力后，每个像素位置都能捕捉所有其他像素的上下文。与非局部模块相比，十字交叉注意力的 GPU 显存用量约为其 1/11，复杂度为 $O(2\sqrt{N})$。文献 [72]

报告了在 Cityscapes [73]、ADE20K [74]、COCO [75]、LIP [76] 和 CamVid [77] 等多个基准数据集上的语义分割与实例分割任务的当时最优结果。

图 5 非局部与十字交叉的注意范围 · 原文 p.5



两种自注意力方法的对比：（a）非局部自注意力模块 [70]；（b）十字交叉自注意力模块 [72]。图来自 [72]。

图中术语 | (a) Non-local block＝非局部模块；(b) Criss-cross attention＝十字交叉注意力。绿色格比较全图与沿行列的注意范围。

卷积算子的另一个不足是，训练完成后，无论视觉输入如何变化，它都施加固定权重。Hu 等人 [78] 提出局部关系网络，自适应地组合局部窗口内的像素。他们引入了一个新的可微层，可依据局部窗口内像素或特征的组合关系（相似性）自适应调整汇聚权重。这种自适应权重汇聚将几何先验引入网络，对识别任务很重要 [78]。卷积在各位置均保持不变，因此被视为一种自顶向下的算子；而文献 [69] 提出的非局部运算是自底向上的方法，因为它会汇聚全图的输入特征。局部关系层也属于自底向上的方法，但局限在固定大小的窗口，例如 7 × 7 邻域。

Bello 等人 [79] 探索以自注意力替代卷积算子的可能性。他们在两个维度上使用相对位置编码 [80]，设计出一种保持平移等变性的新型自注意力机制；这是处理图像时希望具备的特性。尽管该机制作为独立的计算原语就能取得有竞争力的结果，但与卷积运算相结合时表现最好。作者表明，注意力增强能在多种架构中系统性提高图像分类和目标检测的性能。

3.1.2 作为独立计算原语的自注意力

如前所述，卷积层具有平移等变性，但难以扩展到大感受野，因此无法捕捉长程相互作用 [81]。反过来，关注输入所有空间位置的全局注意力 [1] 可能计算代价很高，更适合下采样后的小图像、图像块 [11]，或用于增强卷积的特征空间 [79]。Ramachandran 等人 [81] 提出用局部自注意力层替换深度神经网络中的卷积层；无论输入大小，均可应用该层而不增加计算成本。所提出的自注意力层 [81] 的基本过程是：考察给定像素周围指定大小窗口中的所有像素位置，计算这些像素的查询、键和值向量，再汇聚窗口中的空间信息。对查询与键所得的 softmax 分数做投影后，汇聚值向量。对所有给定像素重复此过程，并拼接响应，得到输出像素。与基于卷积层的 ResNet 相比，带局部自注意力层的 ResNet 使用较少的参数即可完成 ImageNet 任务和 COCO 目标检测 [81]。

Zhao 等人 [82] 指出，传统卷积算子同时完成特征汇聚与变换（施加滤波器后再通过非线性运算）。相比之下，他们提出用自注意力单独做特征汇聚，再用逐元素感知机层做变换。对于特征汇聚，他们提出两种备选策略：(a) 成对自注意力；(b) 逐块自注意力。成对自注意力是对排列与集合基数不变的运算，而逐块自注意力不具备这些不变性，与卷积类似。两者均以向量注意力 [82] 实

现，同时为不同空间位置和不同通道学习权重。这不同于通常用标量权重（通过点积计算）实现注意力的方式。成对自注意力是集合算子；它根据某特征与其局部邻域内各邻居之间的关系计算向量 注意力。相反，逐块自注意力是卷积算子的推广，不是集合算子，在推导注意力向量时考察局部邻 域中的全部特征向量。作者表明，自注意力网络（SAN）可用少得多的参数在 ImageNet 数据集上 超越 ResNet 基线；它还对对抗扰动具有鲁棒性 [83], [84]，并可泛化到未见过的变换 [85]。这是 因为注意力的动态特性让对手难以算出有效的误导方向。

3.2 多头自注意力（Transformer）

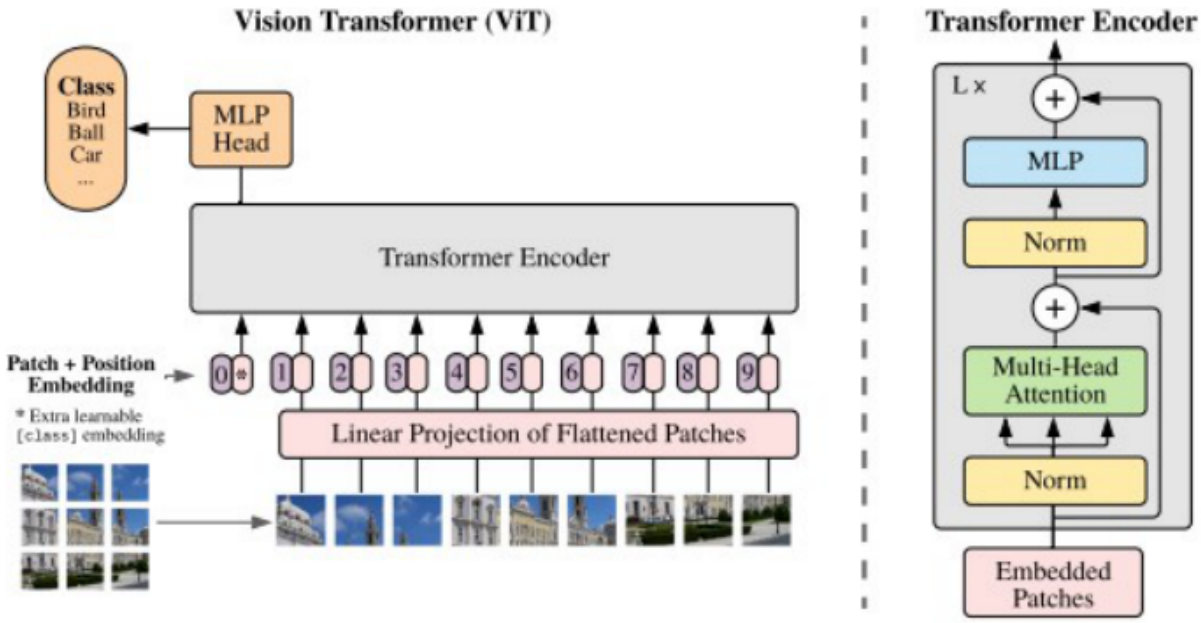
第 3.1 节的方法将自注意力作为组件插入受 CNN 启发的架构，而视觉 Transformer（ViT）[11] 改编了文献 [1] 中级联多个 Transformer 层的架构（见图 3）。ViT 已获得研究界广泛关注，近期出现了不少以它为基础的方法。下文将这些方法分为：各层均采用单一尺度特征的等尺度 ViT（第 3.2.1 节）；学习层级特征、更适合稠密预测任务的多尺度 ViT（第 3.2.2 节）；以及整合了卷积运算的混合 ViT（第 3.2.3 节）。

3.2.1 等尺度视觉 Transformer

原始视觉 Transformer [11] 属于这一类：多头自注意力施加在输入图像中的同一尺度上，并在整个网络层级中保持该空间尺度。我们将这类模型称为等尺度 ViT，如下所述。

视觉 Transformer（ViT）[11]（图 6）率先展示了在大规模图像数据集上，Transformer 如何“完全”替代深度神经网络中的标准卷积。作者将原始 Transformer 模型 [1] 稍作修改，应用于由展平为向量的图像块组成的序列。模型先在大型专有数据集上预训练（含 3 亿张图像，即 300 million 的 JFT 数据集 [47]），再针对 ImageNet 分类等下游识别基准进行微调。预训练是重要的一步，因为在中等规模数据集上预训练 ViT 无法获得有竞争力的结果：CNN 对图像编码了先验知识（归纳偏置，如平移等变性），因而对数据的需求较少；Transformer 则必须从极大规模数据中发现此类信息。值得注意的是，iGPT [19] 同样将 Transformer 用于全尺寸图像，但以生成任务训练；ViT 则以有监督分类任务预训练模型（也探索了自监督变体，但其性能较低）。

图 6 视觉 Transformer（ViT）的结构 · 原文 p.7



视觉 Transformer 概览（左）与 Transformer 编码器的细节（右）。其架构类似 NLP 领域使用的 Transformer，只需将图像块展平后送入模型。训练后，第一个词元位置得到的特征用于分类。图片来自 [11]。

图中术语｜Patch + Position Embedding＝图像块嵌入加位置嵌入；图像块嵌入作为图像词元；Linear Projection of Flattened Patches＝展平图像块后的线性投影；MLP Head＝多层感知机分类头；0* 为可学习的类别嵌入。

DeiT [12] 首次证明，Transformer 可在中等规模数据集上以相对较短的周期训练（ImageNet 的 120 万个样本，即 1.2 million，对比 ViT [11] 使用的 JFT [11] 的 3 亿张图像，即 300 million）。除采用 CNN 常用的数据增强与正则化流程外，DeiT [12] 的主要贡献是一种面向 Transformer 的新型原生蒸馏方法：以一个 CNN（RegNetY-16GF [86]）为教师模型训练 Transformer。CNN 的输出帮助 Transformer 高效找出对输入图像有用的表征。在输入的图像块嵌入和类别词元后再加入一个蒸馏词元。自注意力层对这些词元进行运算，学习相互依赖关系，输出学得的类别词元、图像块词元和蒸馏词元。网络训练采用两项损失：定义在输出类别词元上的交叉熵损失，以及让蒸馏词元匹配教师输出的蒸馏损失。作者探索了软标签和硬标签两种蒸馏方案，结果显示硬蒸馏表现更好。有趣的是，学到的类别词元与蒸馏词元相关性不高，说明两者具有互补性。所学表征与 EfficientNet [87] 等表现最好的 CNN 架构相比颇具竞争力，也能很好地泛化到多种下游识别任务。

Token to Token（T2T）ViT [35] 递归地将相邻词元合成为单个词元，以减少词元序列长度，并汇聚空间上下文。Transformer in Transformer [88] 在两个层级计算注意力：一是与标准 ViT [11] 相同的图像块层级，二是更局部的子图像块层级（比如把一个 16 × 16 图像块分成四个 4 × 4 小块，再计算这些小块之间的注意力）。在采用词元标注的 ViT [89] 中，所有图像块词元都参与损失计算，不像常规 ViT 只用类别词元。该流程还引入了辅助监督：用预训练 CNN 为每个图像块（词元）标注。与 CutMix 增强 [90] 类似，在增强中混合来自不同图像的词元，并用标准分类损失和词元标注辅助损失训练。该模型表现优异，尤其是在模型较小的情况下。

自注意力的二次复杂度限制了它处理较长序列（高分辨率图像）的能力。交叉协方差图像 Transformer（XCiT）[91] 不在词元之间计算注意力，而是在特征通道间计算，其交叉协方差注意力表示为：

$$\mathbf{V}\mathrm{softmax}\left(\frac{\mathbf{K}^T\mathbf{Q}^T}{\sqrt{\tau}}\right).$$

这种交叉协方差注意力的复杂度为线性（因为它依赖特征维度而非词元数量）。因此，XCiT 可以处理高分辨率图像，并在不同视觉任务上表现出色，包括自监督和全监督图像分类，以及检测、分割等稠密预测任务。DeepViT [92] 观察到较深网络层的注意力图相似度很高，阻碍模型深度的扩展。其做法是在多头模块中再次对注意力图施加注意力，而不是简单地汇聚这些图；性能始终优于采用标准多头自注意力的 ViT。

3.2.2 多尺度视觉 Transformer

标准 ViT 在网络各模块中始终维持相同的词元数量和词元特征维度。这限制了模型获取不同尺度精细空间细节的能力，因此早期基于 Transformer 的稠密预测方法（如 DETR [13]）采用卷积后端。研究表明，多阶段层级式 ViT 能产生适合稠密预测任务的有效特征 [36], [93]–[96]：在各阶段逐步减少词元数量，同时逐渐增加词元的特征维度。这些网络通常在识别任务上也表现良好。它们多通过合并相邻词元、将其投影到更高维的特征空间来实现词元稀疏化。多阶段 ViT 包括 Pyramid ViT [93], [97]、Twins [37]、CoaT [98]、Swin Transformer [36]、卷积视觉 Transformer（CvT）[96]、Shuffle Transformer [95]、CrossFormer [99]、RegionViT [100] 和 Focal Transformer [94] 等。其中一些是同时使用卷积和自注意力的混合设计（见第 3.2.3 节），另一些只用自注意力（见下文）。

金字塔 ViT（PVT）[93] 是首个层级式 ViT，提出逐步缩小的金字塔和空间降采样注意力。PVTv2 [97] 和 SegFormer [101] 引入重叠图像块嵌入、逐通道卷积与高效注意力，改进了原版 PVT [93]。Swin Transformer [36] 采用多阶段层级架构，将局部窗口分成多个子图像块，在窗口内部计算注意力。为捕捉不同窗口（图像位置）间的相互作用，网络随层级推进逐步移动窗口的划分位置，从而覆盖相互重叠的区域。另一种层级设计 Focal Transformer [94] 引入焦点自注意力，同时捕捉全局关系与局部关系。类似地，CrossFormer [99] 采用层级金字塔结构，引入跨尺度嵌入模块、长短距离注意力和动态位置偏置，以准确捕捉局部与全局视觉线索。RegionViT [100] 提出从区域到局部的注意力，以编码层级特征。Multi-Scale Vision Longformer [102] 同样考虑自注意力的局部上下文，但所用的注意力机制采用高效的 Longformer [103] 设计。CrossViT [104] 有两个各含多个 Transformer 模块的分支，通过分别处理较大与较小的图像块编码多尺度特征，再用交叉注意力模块融合两个尺度分支的信息。

3.2.3 与卷积结合的混合 ViT

卷积擅长捕捉图像低层局部特征，已被用于多种混合 ViT 架构，尤其用于在网络开头将输入图像“划块并转换成词元”。例如，卷积视觉 Transformer（CvT）[96] 采用基于卷积的投影捕捉空间结构和低层细节，用来将图像块转换成词元。CvT 采用层级设计，逐步减少词元数量、增大词元宽度，从而仿照 CNN 的空间下采样效果。卷积增强的图像 Transformer [105] 使用基于卷积的图像到词元模块提取低层特征。紧凑卷积 Transformer（CCT）[106] 引入新的序列池化方案，并加入用于生成词元的卷积模块（卷积–池化–重塑）。CCT 可以在 CIFAR10 等较小数据集上从头训练，准确率约为 95%，而传统 ViT 无法做到这一点。

LocalViT [107] 引入逐通道卷积，增强 ViT 对局部特征的建模能力。LeViT [108]（名称来自 LeNet [109]）在开头使用一个四层 CNN 模块（使用 3 × 3 卷积），逐渐增加通道数（3, 32, 64, 128, 256）。输入 3 × 224 × 224 的图像后，该 CNN 模块生成的 256 × 14 × 14 输出成为层级式 ViT 的输入。凭借这一设计，LeViT 在 CPU 上的推理速度是 EfficientNet [87] 的 5 倍。ResT [110] 是另一种层级架构，也在开头应用了一个 CNN 模块进行图像块嵌入。它使用逐通道卷积和自适应位置编码处理不同尺寸的图像。近期的 NesT [111] 用一种简单的方法给 ViT 引入层级：先将图像划分为互不重叠的大块（每个大块再切分为图像块），然后分别对各大块内的图像块施加局部自注意力；再将各大块汇聚回图像空间，经过卷积与下采样，使大块之间发生全局交互。随着模型层级加深，大块的数量逐步减少，而局部图像块的数量保持不变。这一简单方案与更复杂的设计 [36], [97] 相比表现良好，且能在 CIFAR-10 等较小数据集上从头训练 NesT。

逐通道卷积与自注意力网络（CoAtNets）[112] 引入一种结合逐通道卷积与自注意力的相对注意力模块，并沿网络深度堆叠卷积层与注意力层。在没有额外数据（即只在 ImageNet-1k 上训练）的条件下，CoAtNets 在 ImageNet 上取得了 86% 的 Top-1 准确率。Shuffle Transformer [95] 在窗口内进行自注意力，在基于窗口的多头自注意力与 MLP 之间加入逐通道卷积，并引入重排操作加强跨图像块连接。协同尺度卷积注意力图像 Transformer（CoaT）[98] 是包含串行和并行模块的混合层级金字塔。串行模块类似标准 Transformer 模块，但以逐通道卷积替代注意力层；并行模块作用于串行模块输出，通过交叉注意力编码不同尺度词元间的关系。Twins [37] 以仅采用注意力的金字塔架构 PVT [93] 为基础，将其中的绝对位置嵌入换为相对条件位置嵌入 [113]，并以可分离的逐通道卷积替代标准空间注意力，来捕捉图像的局部与全局上下文。这类混合设计试图结合卷积模型和 Transformer 模型各自的优势。TransCNN [114] 提出层级多头自注意力模块，先用自注意力学习小网格（词元）内的相互作用，再逐步将小网格合并为大网格。所提模块可接入现有 CNN 架构。

3.2.4 自监督视觉 Transformer

基于对比学习的自监督方法已在 CNN 视觉任务上取得显著成效，也被用于 ViT。Chen 等人 [115] 评估了不同的自监督框架，提出 MoCo v3（源于 v1/v2 [116], [117]）等实用策略，以稳定自监督 ViT 的训练。Xie 等人 [118] 结合 MoCo v2 [117] 和 BYOL [119]，训练 DeiT [12] 及 SwinTransformer [36]，并表明自监督 SwinTransformer 能泛化到检测和分割等稠密预测任务。无标签自蒸馏（DINO）[120] 表明，即使训练时从未使用像素级监督，自监督 ViT 也能自动分割图像的背景像素，而 CNN 和全监督 ViT 中没有观察到这种现象。高效自监督视觉 Transformer（EsViT）[121] 提出多阶段设计，在网络层级间逐步合并相邻词元，并采用 DINO 进行自监督。除 DINO 的标准图像级自监督外，它还增加了图像块级自监督，使同一图像不同增强版本中相似的图像块建立对应关系。EsViT 在自监督条件下表现优异；在评估的 18 个数据集中的 17 个上，其直接提取的特征具有比有监督 SwinTransformer 更好的迁移效果。

3.3 用于目标检测的 Transformer

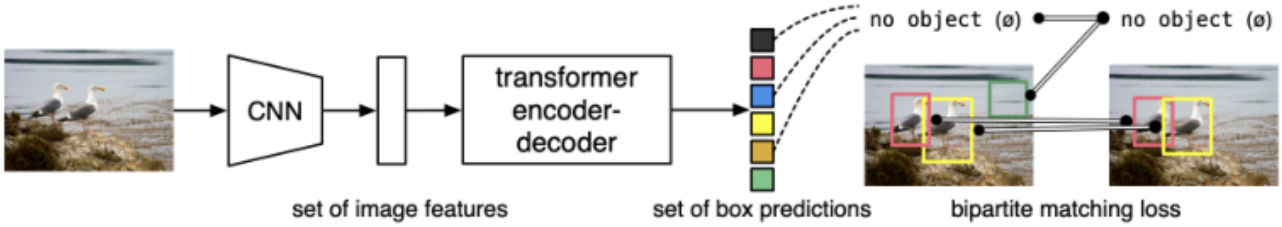
基于 Transformer 的模块已通过以下方式用于目标检测：(a) 以 Transformer 为骨干网络提取特征，用基于 R-CNN 的头部做检测（见第 3.2.2 节）；(b) 以 CNN 骨干网络提取视觉特征，用基于 Transformer 的解码器做目标检测 [13], [14], [122], [123]（见第 3.3.1 节）；(c) 用纯 Transformer 架构做端到端目标检测 [124]（见第 3.3.2 节）。

3.3.1 使用 CNN 骨干网络的检测 Transformer

检测 Transformer（DETR）[13] 将目标检测视为集合预测：给定一组图像特征，目标是预测目标边界框的集合。Transformer 模型能够一次预测一组目标，并对其相互关系建模。DETR 采用一种集合损失函数，能在预测框和真值框之间做二分匹配。DETR 的主要优势是不再依赖手工设计的模块和运算，例如目标检测中常用的 RPN（区域提议网络）和 NMS（非极大值抑制）[125]–[129]。这样一来，对于目标检测这类复杂的结构化任务，模型对先验知识和精细工程设计的依赖有所减轻。

给定 CNN 骨干网络生成的空间特征图，编码器首先展平空间维度（见图 7），得到形状为 d × n 的特征序列，其中 d 是特征维度，n = h × w，h 和 w 分别为空间特征图的高和宽。之后，如文献 [1] 所述，使用多头自注意力模块对这些特征进行编码与解码。解码阶段的主要差别是：所有框同时预测，而文献 [1] 使用 RNN 逐个预测序列元素。由于编码器和解码器具有排列不变性，解码器使用学到的位置编码作为目标查询，生成不同的框。CNN 检测器（如 Faster R-CNN）则通过空间结构自动编码位置信息。设计简明的 DETR 取得了与常用的 Faster R-CNN [125] 相当的性能。DETR 也已扩展到其他领域，例如 Cell-DETR [130] 扩展它以实现生物细胞的实例分割。除了预测边界框，还新增一个专门的注意力分支输出逐实例的分割结果，再用 CNN 解码器增强，以得到准确的实例掩码。

图 7 DETR 的集合预测和二分匹配 · 原文 p.9



检测 Transformer（DETR）[13] 把目标检测视为集合预测问题，用 Transformer 网络对集合元素之间的关系编码。通过二分集合损失，将预测框与真值框唯一匹配（最右两列）。没有匹配时，预测为“无目标”类。这个架构设计简单，只需针对任务作很少的修改，即可超过精心设计且广为使用的 Faster R-CNN。图片来自 [13]。

图中术语｜Set of image features＝图像特征集合；Set of box predictions＝预测框集合；Bipartite matching loss＝二分匹配损失；No object＝无目标。

DETR [13] 成功结合了卷积网络与 Transformer [1]，免去手工设计组件，实现了可端到端训练的目标检测流程。但它难以检测小目标，收敛也较慢，计算成本相对较高 [14]。在用 Transformer 对关系建模前，DETR 先将图像映射到特征空间，所以自注意力计算成本随特征图空间尺寸呈二次增长，即 $O(H^2W^2C)$，其中 H 和 W 分别表示特征图的高和宽。这限制了 DETR 训练时使用多尺度层级特征 [131]，而此类特征对检测小目标很重要。并且，训练初期，注意力模块对特征图的各个位置施加均匀的注意力，需要大量训练轮次才能把注意力权重收敛到有意义的少数位置。这也是 DETR 收敛较慢的原因之一。为缓解上述问题，文献 [14] 提出以可变形注意力模块处理特征图。受可变形卷积 [42] 启发，无论特征图有多大，可变形注意力模块 [14] 都只关注图中特定的少量元素。利用多尺度注意力模块，还能在不显著增加计算成本的情况下跨尺度汇聚特征图。Deformable DETR 不仅性能更好，训练时间也只有原始 DETR 模型的约 1/10 [14]。Anchor DETR [122] 将文献 [13] 中的可学习查询词元换成基于锚点的查询，使每个查询集中预测锚点附近的目标。锚点可以固定在二维网格上，也可以从均匀分布的点上学习得到。Anchor DETR [122]

仅用约 1/10 的训练轮次即可取得相当的性能。Pix2Seq [123] 是一种通用的 Transformer 框架，没有专门面向任务的模块，学习直接生成一串描述目标的词元（边界框和类别标签）。先用量化和 序列化把边界框和类别标签转成离散词元序列，再用通用 Transformer 编码器–解码器网络，以此 前的预测和图像特征为条件，自回归地生成这些词元。

3.3.2 采用纯 Transformer 的检测

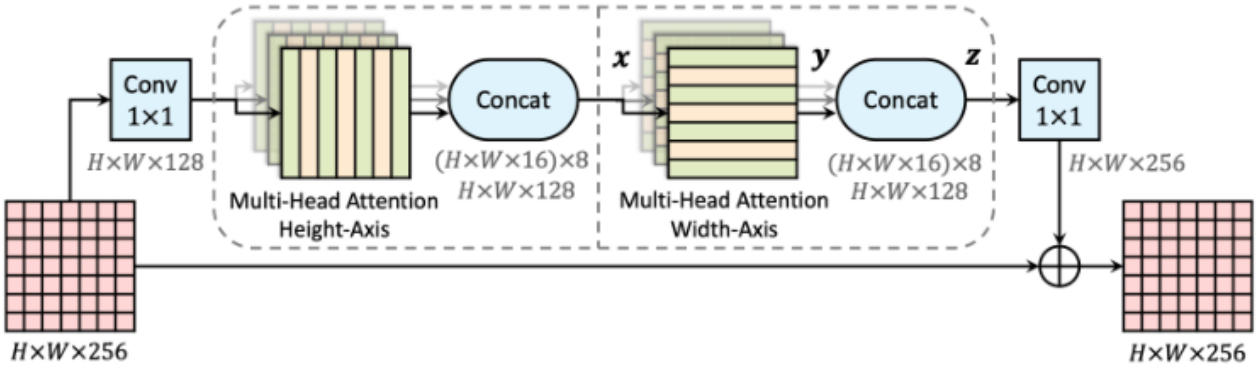
You Only Look at One Sequence（YOLOS）[124] 是直接建立在 ViT [1], [132] 上的简洁纯注意力架构。它用多个可学习的目标查询词元替换 ViT 的类别词元，并像文献 [13] 一样用二分匹配损失进行目标检测。YOLOS 表明，ViT 能灵活适应目标检测，只需极少与二维图像相关的归纳偏置，就能以纯粹的序列到序列方式学习。基于类似思想，PVT [93] 与 DETR [13] 结合，形成端到端的 Transformer 目标检测流程。我们注意到，其他近期 ViT 也可与基于 Transformer 的检测头结合，构成只使用 ViT 的设计 [124]，希望将来看到更多此类工作。

3.4 用于分割的 Transformer

自注意力可用于图像分割等稠密预测任务，这些任务需要对像素之间丰富的相互作用建模。下文讨论轴向自注意力 [133]、可根据给定语言表达分割对应区域的跨模态方法 [15]，以及基于 ViT 的分割架构 [101], [134], [135]。

全景分割 [136] 旨在联合解决原本不同的语义分割和实例分割两项任务，为每个像素指派语义标签和实例 ID。全局上下文可为这一复杂的视觉理解任务提供有用线索。自注意力能有效建模长程上下文信息，但在全景分割之类的稠密预测任务中，把它施加到大尺寸输入上的代价往往过高。一个直接办法是只在下采样后的输入或各像素附近有限的区域中使用自注意力 [81]。即使引入这些限制，自注意力仍存在二次复杂度，还会牺牲全局上下文。为解决这些问题，Wang 等人 [133] 提出位置敏感的轴向注意力，把二维自注意力重写为两个一维轴向注意力层，分别依次作用于高度轴与宽度轴（见图 8）。轴向注意力计算效率较高，并使模型能够捕捉全图上下文。它在 COCO [75]、Mapillary Vistas [137] 和 Cityscapes [73] 的全景分割基准以及 ImageNet [138] 的图像分类上均获得有竞争力的性能。

图 8 轴向注意力 · 原文 p.9



轴向注意力模块 [133]，沿高度轴和宽度轴依次应用多头轴向注意力运算。图片来自 [133]。

图中术语 | Multi-Head Attention Height/Width-Axis＝沿高度/宽度轴的多头注意力；Concat＝拼接；Conv 1×1＝1×1 卷积。

跨模态自注意力（CMSA）[15] 为指代表达图像分割任务编码语言特征与视觉特征之间的长程跨模态依赖；该任务旨在分割图像中由语言描述指代的实体。为此，将图像特征与每个词的嵌入及空间坐标特征拼接，得到一组跨模态特征。自注意力作用于这些特征，生成与句中各词相对应的图像注意力。分割网络随后在多个空间层级上执行自注意力，并用门控的多层级融合模块在不同分辨率的特征之间交换信息，细化分割掩码。整体模型用二元交叉熵损失训练，在 UNC [139]、G-Ref [140] 和 ReferIt [141] 数据集上取得了良好的提升。

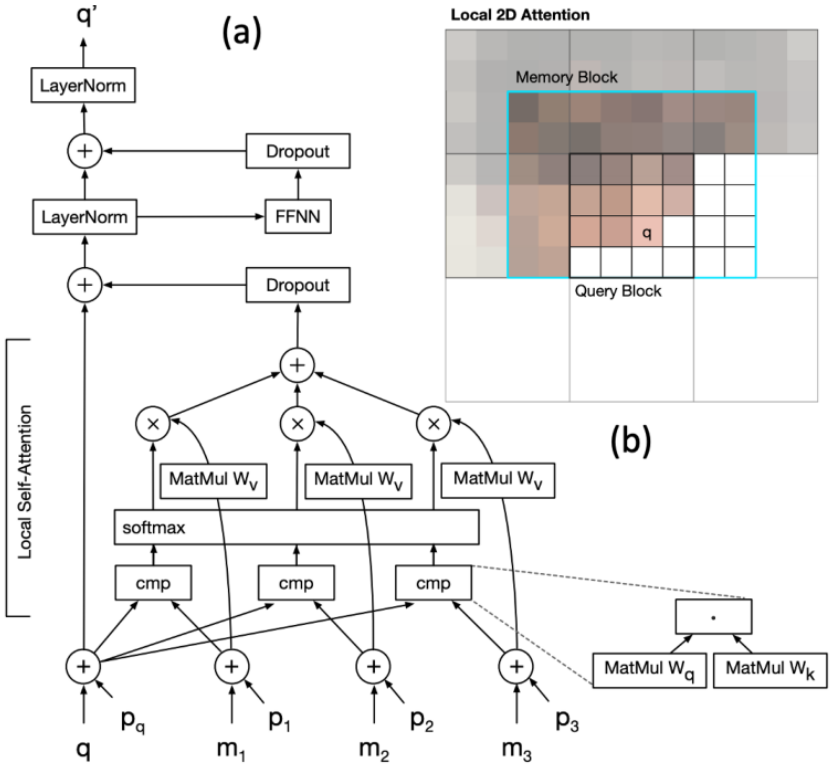
以上方法在 CNN 架构中嵌入自注意力；而近期一些工作提出基于 Transformer 的编码器–解码器架构。分割 Transformer（SETR）[134] 使用 ViT 编码器，并有两种解码器设计，分别基于渐进上采样与多层特征汇聚。SegFormer [101] 使用无位置编码的层级金字塔 ViT [93] 作为编码器，用简单的 MLP 解码器配合上采样得到分割掩码。Segmenter [135] 用 ViT 编码器提取图像特征，解码端采用掩码 Transformer 模块，以可学习的掩码词元和图像块词元为输入，预测分割掩码。作者还提出一个线性解码器基线，将图像块嵌入投影到分类空间，生成较粗的图像块级标签。

3.5 用于图像和场景生成的 Transformer

这里讨论基于 Transformer 的图像合成架构 [23], [142]–[146]。从生成建模和为下游任务学习无监督表征的角度来看，这些架构值得关注。

Parmar 等人 [142] 开发了一种图像生成模型，可以根据此前生成的像素，顺序预测输出图像中的每个像素（图 9）。该方法把图像像素的联合分布分解成逐像素条件分布的乘积。此前为此任务开发的自回归模型，例如 PixelCNN [147]，感受野有限，难以建模图像中的长程关系，如部件关系或遮挡。利用自注意力，文献 [142] 扩大了感受野而不引入高昂的计算代价，例如有效感受野可达 256 像素，而 PixelCNN [147] 只有 25 像素。该生成流程也在图像超分辨率、图像补全和去噪等条件生成任务上进行了测试。

图 9 局部自注意力生成图像 · 原文 p.10



(a) Image Transformer [142] 中的自注意力模块。给定像素 q 的一个通道，该模块关注先前已经合成的像素记忆（m_i），随后经过前馈子网络。第一层加入位置编码 p_i。(b) 局部自注意力的运算示例（二维情形）。图像被划分成空间块网格，这些块称为查询块。执行自注意力时，查询块中的每个像素都会关注记忆块（青色矩形）中的所有像素。网格中的白色位置表示被掩蔽的输入，对自注意力没有贡献。

图中术语｜Local 2D Attention＝局部二维注意力；Memory Block＝记忆块；Query Block＝查询块；LayerNorm＝层归一化；FFNN＝前馈神经网络。

受 GPT [5] 在语言领域成功的启发，图像 GPT（iGPT）[143] 表明，这类模型可以直接用于图像生成，也可以学习适用于图像分类等下游视觉任务的强特征。具体而言，iGPT 用展平的图像序列（一维像素数组）训练 GPT v2 [5]，展现出在没有任何外部监督时生成合理图像的能力。所生成的样本说明模型能够理解像素间的空间关系，以及目标类别、纹理和尺度等高层属性。值得注意的是，该设计未使用图像特有的知识，例如 Image Transformer [142] 中的二维位置嵌入。iGPT 用无监督训练机制学到的特征与其他无监督方法相比颇具竞争力：它在 CIFAR-10/100 [148] 和 STL [149] 上达到当时最优水平，在 ImageNet 上表现与对比学习方法 SimCLR [150] 相当。这一结果令人惊讶，因为 iGPT 架构与语言建模中使用的架构完全相同，因而没有引入任何领域先验。相比之下，与之竞争的无监督 CNN 方法广泛以架构设计、注意力机制、损失函数或正则化等形式引入这些先验 [117], [151]–[154]。但 iGPT 的缺点是计算成本高。例如，iGPT-L 的训练成本约为当时领先的自监督特征学习方法 MoCo [117] 的 36 倍。因此，其训练一般限于不超过 64 × 64 的低分辨率，而卷积架构可以有效地从高分辨率输入学习。

Transformer 处理高维序列时往往计算开销较高。为克服这一限制，Esser 等人 [144] 提议在 Transformer 中结合常用于 CNN 的归纳偏置，以提高效率。具体来说，利用 CNN 结构内置的局部连接和空间不变性偏置，通过生成对抗方法学习一个丰富的视觉模式字典；再用 Transformer 学习字典各项之间的长程相互作用，以生成输出。由此构建了能够使用 Transformer 生成超高分

辨率图像（达百万像素量级）的条件图像生成模型。这是首次展示 Transformer 可用于生成如此高分辨率的图像。

默认采用 CNN 骨干网络的生成对抗网络（GAN）[54] 在合成视觉效果良好的图像方面已经非常成功 [155]–[157]。TransGAN [145] 构建了一个性能强、不含任何卷积运算的 GAN，生成器和判别器均基于 Transformer [1]。两者的架构都基于原始 Transformer [1] 的编码器。为提高内存效率，生成器包含多个阶段，并在其间加入上采样模块，逐步提高特征图分辨率（输入序列长度），同时降低嵌入维度。TransGAN 的判别器将展平的图像块作为词元，类似于文献 [132]。为扩展模型以合成高质量图像 [144]，作者引入了多种训练技术，包括数据增强、用辅助任务训练以及向自注意力注入局部性。TransGAN 在 STL-10 上的 Inception Score 和 Fréchet Inception Distance（FID）达到当时最优，与其他数据集上同类的 CNN 型 GAN 相比，表现也较好。

与直接预测图像输出的早期图像生成方法 [142]–[144] 不同，文献 [23] 学习生成待放置于给定场景中的三维物体参数。具体来说，SceneFormer [23] 研究以房间布局为条件的三维场景生成。给定空房间的形状，文献 [23] 的方法能在保持真实感的前提下提出新的室内物体布置。值得注意的是，该模型不使用任何外观信息，仅通过 Transformer 的自注意力对物体间关系建模来学习生成新场景。类似于在句子上处理词序列，模型在物体序列上运算，预测场景中下一个合适的物体。具体预测下一个物体的尺寸、姿态、位置和类别。起始词元表示开始推理，输出词元的数量对应模型顺序生成的物体。作者还探索了根据房间布局的文本描述生成新场景。不依赖外观信息使该方法效率较高，从而能够交互式生成场景。

从文本生成逼真的图像既有趣、又有实际价值（例如用于艺术内容创作），同时也很有挑战。早期文本到图像合成方法 [158]–[161] 多基于 GAN [54]。虽然这些方法得到的结果令人鼓舞，但离照片级真实仍有很大距离。Ramesh 等人 [20] 提出的 DALL • E 是一个可以根据给定文本描述生成高保真图像的 Transformer 模型。DALL • E 拥有 120 亿个参数（12 billion），在互联网上收集的大量文本–图像对上训练。训练前，先将图像调整到 256 × 256 分辨率，再用预训练的离散变分自编码器 [162], [163] 压缩为 32 × 32 的潜变量编码网格。DALL • E 的输入为单一词元流，共 1280 个词元（文本占 256 个，图像占 1024 个），其训练目标是自回归地逐个生成其余词元。它既可从头生成图像（图 10a），也可扩展现有图像（图 10b），并忠实于文本描述。

作者根据描述多种真实与虚构概念的文本生成图像，以展示 DALL • E 的有效性。仅从文本描述生成图像时，DALL • E 在控制多个物体及其属性（图 10c）、渲染特定视角（图 10d）、表现物体内部结构（图 10e）和组合不相关的物体（图 10f）方面表现出色。此外，DALL • E 还可在输入文本引导下实现图像到图像的转换（图 10g）。

图 10 DALL·E 的生成、补全与图像转换 · 原文 p.12



(a) (b) (c) (d) (e) (f) (g)

DALL • E [20] 根据以下文字提示生成的图像。(a) 一把牛油果造型的扶手椅。(b) 旧金山金门大桥的一张照片。给定图像的一部分（绿框内），DALL • E 补全图像。(c) 戴蓝帽子、红手套，穿绿衬衫和黄裤子的企鹅宝宝表情图。(d) 一只坐在田野里的水豚的极近特写。(e) 石榴的剖面图。(f) 一只由西瓜做成的企鹅。(g) 将上方同一只猫画成下方的素描。

图中术语 | 按原图 (a)–(g) 顺序对应图注。图 (b) 的绿色框标出补全任务给定的原始图像区域。

3.6 用于低层视觉的 Transformer

看到 Transformer 在高层视觉问题中的成功后，研究者针对低层视觉任务提出了许多基于 Transformer 的方法，包括图像超分辨率 [16], [19], [164]、去噪 [19], [165]、去雨 [19], [165] 和上色 [24]。图像复原要求输入和输出的像素一一对应。复原算法的一个主要目标是保留图像中所需的精细细节，如边缘和纹理。CNN 通过不包含下采样的单尺度架构实现这一点。由于 Transformer 自注意力的计算复杂度随图像块数量呈二次增长，开发按单尺度处理特征的 Transformer 模型并不现实。因此，基于 Transformer 的图像复原模型采用多种策略减轻计算负担，例如在局部图像窗口内计算注意力 [164]、采用空间降采样注意力 [166] 以及采用编码器–解码器设计 [19], [165]。这里简要讨论几个用于图像复原的 Transformer 模型。

3.6.1 用于图像处理任务的 Transformer

用于目标检测和语义分割等高层计算机视觉任务的顶尖算法，通常采用在 ImageNet 等大规模数据集上预训练的骨干网络。相比之下，图像去噪、超分辨率及去雨等低层视觉任务的算法直接使用特定于任务的数据进行训练，因此面临两项局限：(i) 特定任务数据集的图像数量较少，例如超分辨率中常用的 DIV2K 只有 2000 张图像；(ii) 针对一个图像处理任务训练的模型不容易适应其他相关任务。

Chen 等人 [19] 提出以 Transformer 架构为基础的预训练模型，称为图像处理 Transformer（IPT）。它能够执行超分辨率、去噪、去雨等多种图像复原任务。IPT 的整体架构由多个分别面向不同任务的输入头与输出尾部，以及一个共享的 Transformer 编码器–解码器主体构成。充分发挥 Transformer 的潜力需利用大规模数据训练，因此文献 [19] 从 ImageNet 基准中选取干净的图像作为真值，并针对不同任务合成相应的退化图像。例如，用双三次插值生成低分辨率图像、加入加性高斯白噪声得到有噪数据、叠加手工设计的雨纹生成雨图。IPT 预训练共使用了 1000 万张图像（10 million）。训练中，各任务专用的输入头接收退化图像，生成视觉特征。将这些特征图裁成小块，随后展平并输入 Transformer 编码器（架构与文献 [1] 相同）。编码器输出与任务嵌入一同送入 Transformer 解码器。将解码器输出的特征重塑后，送入相应的输出尾部，得到复原图像。IPT 使用 L1 损失优化。实验表明，将预训练 IPT 针对某一低层视觉任务微调后，可显著优于当时最先进的方法 [167]–[169]。

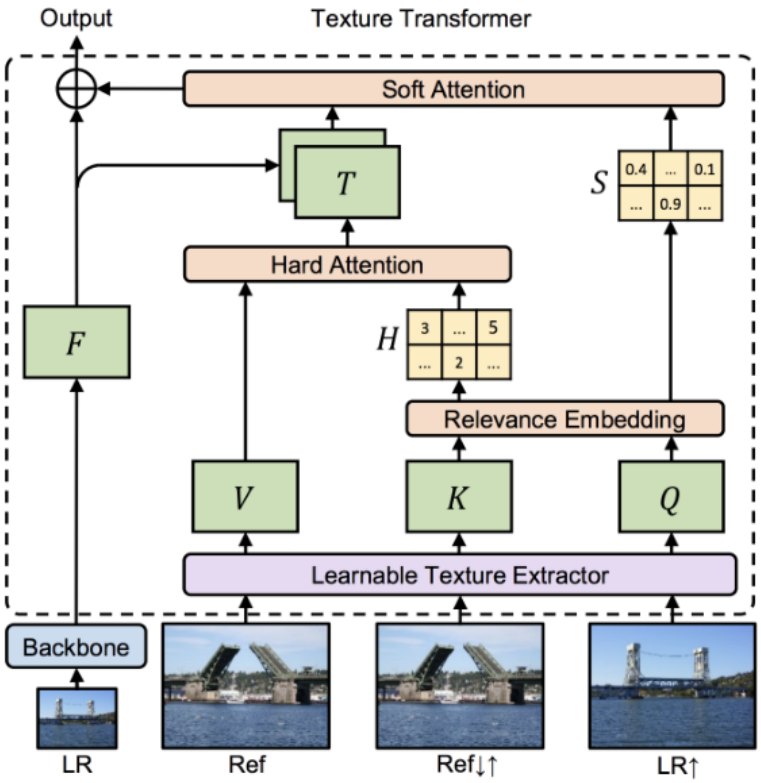
3.6.2 用于超分辨率的 Transformer

近年来，卷积神经网络（CNN）推动超分辨率（SR）性能取得重大突破。原则上，CNN 生成的超分辨率图像的质量取决于优化目标。基于逐像素损失（例如 L1、MSE）的 SR 方法 [167], [170]–[173] 在 PSNR、SSIM 等图像保真度指标上取得了出色结果，却难以恢复精细纹理，且常生成过于平滑、观感欠佳的图像。另一方面，感知型 SR 方法 [52], [174]–[177] 除逐像素损失外，还使用对抗损失 [54]，以及基于预训练 CNN 深层特征的感知损失 [178]。这些方法能产生清晰、赏心悦目且符合感知的图像，但按 PSNR/SSIM 衡量的重建准确度大幅下降。此外，感知型 SR 算法倾向于凭空生成虚假纹理，造成伪影。上述 SR 方法遵循两个不同且相互矛盾的研究方向：一是最大化重建精度，二是最大化感知质量，而无法兼顾两者。

为缓解感知保真与精确重建之间的取舍，Yang 等人 [16] 提出用于超分辨率的 Transformer 网络 TTSR。训练时，TTSR 使用配对的低分辨率（LR）与高分辨率（HR）图像，也使用内容与 LR 图像相似的参考图像（Ref）。TTSR 学习在 Ref 中找出相关区域，并迁移丰富的纹理，帮助提高输入 LR 图像的分辨率。TTSR 的纹理 Transformer 模块（见图 11）包含四个核心组件：(1) 可学习的纹理提取器：接收 LR↑、Ref↓↑ 和 Ref 图像，分别生成作为查询（Q）、键（K）和值（V）的纹理特征。这里 ↑ 表示双三次上采样，↓↑ 表示双三次下采样后再上采样。(2) 相关性嵌入：先将 Q 和 K 展开成图像块，再计算 Q 中各块与 K 中各块的相似度，生成硬注意力图与软注意力图。(3) 硬注意力：用硬注意力图把 HR 纹理特征从 V 迁移到 Q（LR 特征）。(4) 软注意力：进一步增强相关特征，并抑制相关性较低的特征。

TTSR [16] 针对有参考图像的超分辨率，而大多数研究针对单图像超分辨率问题，只有配对的 LR–HR 图像可用。由于原始自注意力在高分辨率图像上计算复杂度过高，近期提出了若干高效的 Transformer 超分辨率模型，例如采用窗口注意力的 SwinIR [164]，以及在注意力模块中降低空间分辨率的 ESRT [166]。

图 11 TTSR 的纹理 Transformer · 原文 p.13



纹理 Transformer 模块示意。Q（查询）、K（键）和 V（值）分别表示从经双三次上采样的低分辨率图像、先下采样再上采样的参考图像，以及原始参考图像中提取的纹理特征。相关性嵌入用于估计低分辨率图像与参考图像之间的相似度。H 和 S 分别表示由相关性嵌入计算出的硬注意力和软注意力。T 表示高分辨率纹理特征，随后将其迁移到低分辨率图像的特征 F。图片来自 [16]。

图中术语｜Learnable Texture Extractor＝可学习纹理提取器；Relevance Embedding＝相关性嵌入；Hard/Soft Attention＝硬/软注意力；Backbone＝骨干网络。

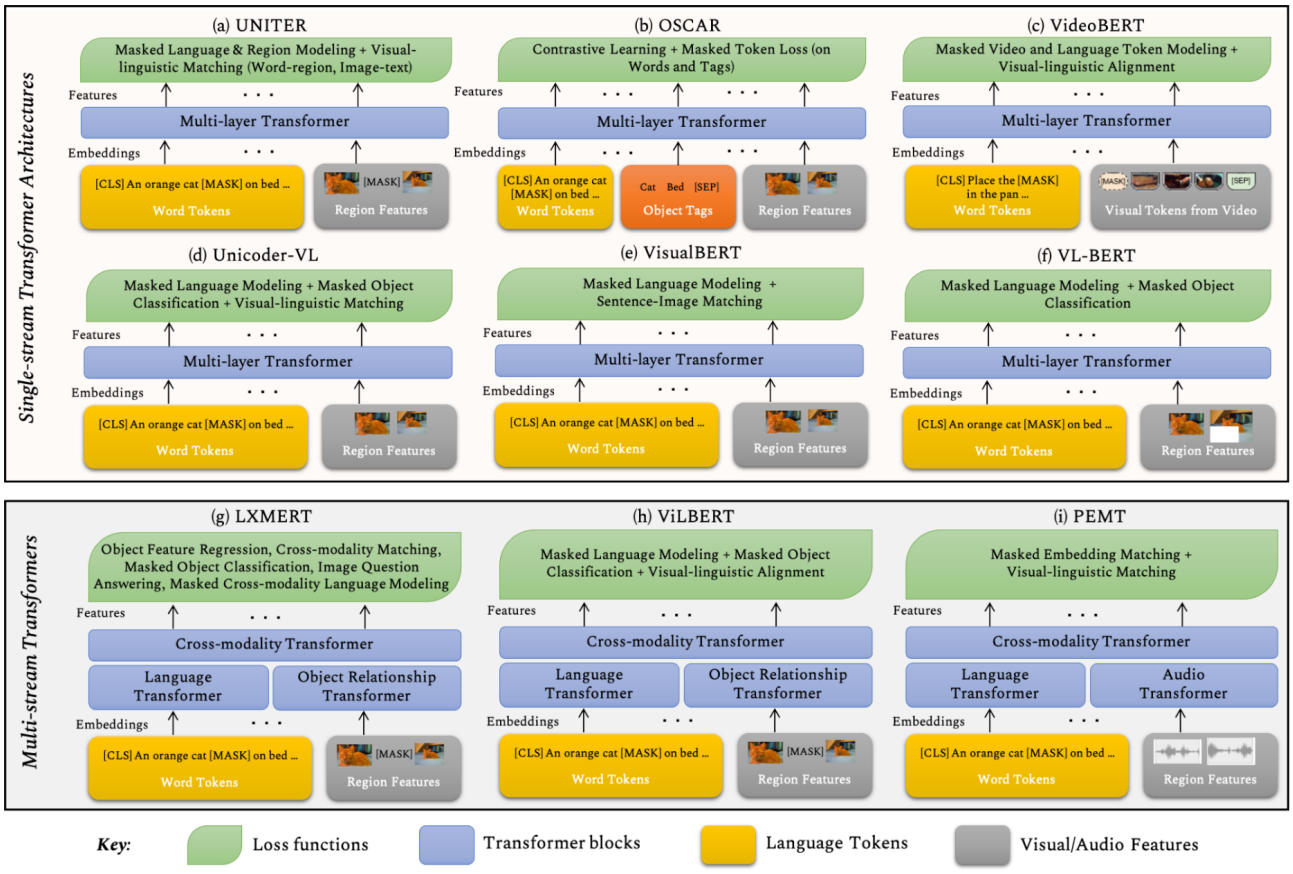
3.6.3 上色 Transformer

图像上色的目标是给定灰度图像，生成相应的彩色样本。这是一对多的任务，因为给定同一灰度输入，彩色输出空间中有多种可能解。为应对这一挑战，需要能生成多个彩色输出样本的概率模型。Colorization Transformer [24] 是一种基于条件注意力机制 [179] 的概率模型。它把上色任务划分为三个子问题，依次用不同的 Transformer 网络求解。作者首先训练一个 Transformer，将低分辨率灰度图映射到 3 位的低分辨率彩色图；采用低分辨率图像可以训练更大的模型。在训练的第二阶段，用另一个 Transformer 将 3 位的低分辨率彩色图提升为 8 位 RGB 样本。最后训练第三阶段的 Transformer，提升第二阶段输出的 8 位 RGB 样本的空间分辨率。上色 Transformer 的自注意力基于文献 [179] 引入的行/列注意力层。这些层以较低的计算代价捕捉输入图像中各像素之间的相互作用。行注意力层对给定行的所有像素施加自注意力，列注意力层只考察给定列的像素。这项工作 [24] 首次成功地训练 Transformer 为高分辨率（256 × 256）的灰度图像上色。

3.7 用于多模态任务的 Transformer

Transformer 模型也广泛用于视觉–语言任务，比如视觉问答（VQA）[183]、视觉常识推理（VSR，按原文记号）[184]、跨模态检索 [185] 和图像描述生成 [29]。这一方向的若干工作致力于在大规模多模态数据集上进行有效的视觉–语言预训练（VLP），学习能编码跨模态关系的通用表征，例如在给定图像中定位某人的语义属性。之后可将这些表征迁移到下游任务上，往往取得当时最优结果。值得注意的是，其中一些模型仍用 CNN 作视觉骨干网络提取视觉特征，Transformer 主要用于编码文本，随后融合语言特征与视觉特征。这些方法通常将多模态输入交给标准的多层 Transformer [1]，不对核心注意力模块作根本改动，主要差别在于 Transformer 的配置和损失函数。据此可分为 (a) 多流 Transformer（见第 3.7.1 节）和 (b) 单流 Transformer（见第 3.7.2 节）。单流设计将多模态输入送入一个 Transformer；多流设计先为每种模态分别使用独立的 Transformer，而后用另一个 Transformer 学习跨模态表征（见图 12）。除视觉–语言预训练方法外，本节末还介绍视觉指代定位方法（见第 3.7.3 节）。

图 12 单流与双流多模态架构 · 原文 p.14



计算机视觉多模态任务所用的 Transformer 模型概览。这类设计可分为单流架构（UNITER [43]、OSCAR [44]、VideoBERT [17]、Unicoder-VL [180]、VisualBERT [63]、VL-BERT [22]）和双流架构（LXMERT [21]、ViLBERT [181]、PEMT [182]）。损失函数的选择是模型之间的一项主要差别。多数多模态方法以图像作为视觉数据，而 VideoBERT [17] 和 PEMT [182] 面向视频流，可利用视频中的音频信号等特有模态 [182]。

图中术语｜绿框＝损失函数；蓝框＝Transformer 模块；黄框＝语言词元；灰框＝视觉或音频特征。(a)–(f) 为单流，(g)–(i) 为双流。Cross-modality Transformer＝跨模态 Transformer；Masked Language Modeling＝掩码语言建模；Visual-Linguistic Matching＝视觉–语言匹配。

3.7.1 多流 Transformer

视觉与语言 BERT（ViLBERT）[63] 是 BERT 模型向多模态领域的首次扩展，目标是学习能够联合建模图像和自然语言的表征。为此，ViLBERT 设计了双流架构，每一流专门处理视觉或语言输入（图 12-h）。两个并行流的架构都是一系列类似 BERT 的 Transformer 模块，随后通过协同注意力 Transformer 层学习跨模态关系。协同注意力框架很简单：按标准方式 [1] 分别为每种模态计算查询、键和值矩阵，再把一种模态的键–值对交给另一种模态的注意力头。

ViLBERT 基于 Conceptual Concepts 数据集（330 万张，即 3.3M 张带弱监督描述的图像）定义若干代理任务进行 VLP，之后在视觉问答等下游任务上微调。预训练以自监督方式开展，即在大规模无标签数据集上创建不依赖人工标注的代理任务，既预测文本与图像输入是否相关，也预测被掩蔽图像区域及文本输入的语义（类似于 BERT [3] 重建文本中的被掩蔽词）。这样，模型在预训练中学到数据的内在结构，同时建模跨领域联系。在若干任务上评估后，文献 [17] 表明，双流模型的表现可以优于用共享参数建模视觉和语言两领域的单流模型 [17]。

与 ViLBERT [181] 类似，基于 Transformer 学习跨模态编码器表征的 LXMERT [21] 也采用基于 BERT 的双流架构。主要区别是，LXMERT 使用目标关系编码器对视觉特征建模，而不是使用 ViLBERT 的简单图像级特征。随后，与文献 [181] 类似，使用交叉注意力模块跨模态融合两个流的信息。

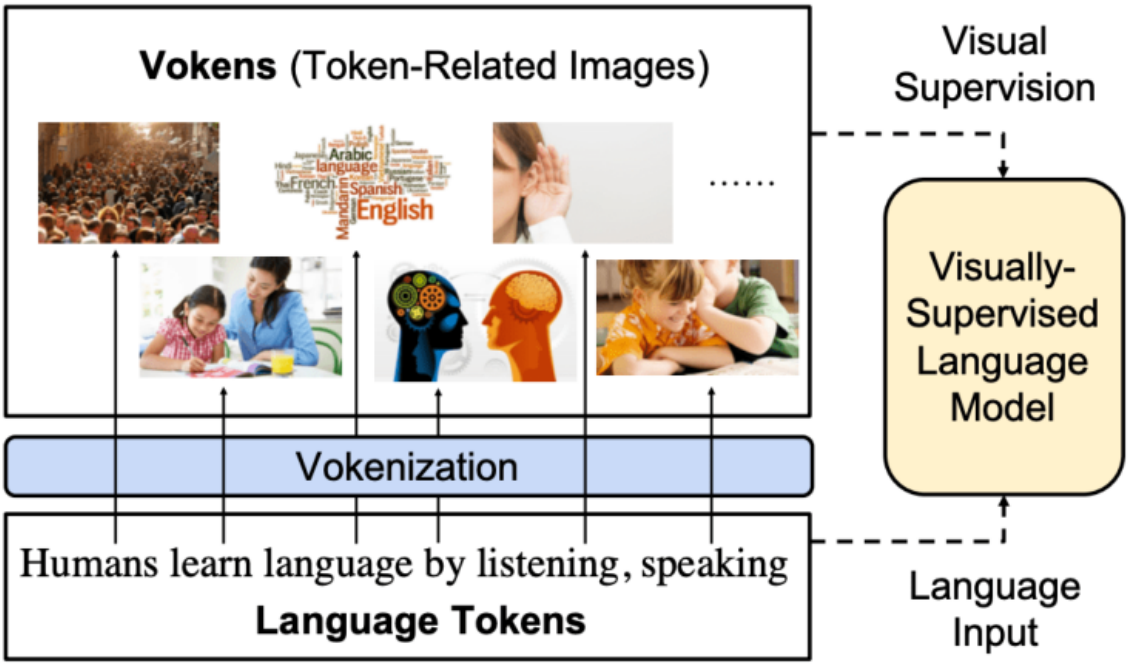
与文献 [181] 为 VLP 使用的两个代理任务不同，LXMERT 使用五项预训练任务，包括掩码目标预测和掩码语言预测、跨模态匹配及视觉问答（图 12-g）。预训练模型针对 VQA 任务微调，但预训练与微调任务高度相似，这让人质疑学到的表征能否泛化到新任务。因此，作者在真实视觉推理（NLVR）任务 [186] 上做了泛化实验，显示在新任务上有显著提升。

Lee 等人 [182] 指出，VideoBERT [17]、ViLBERT [181] 等多模态表征学习方法一般将语言处理部分固定为一个预训练模型（如 BERT [3]），以降低训练复杂度。他们首次提出从无标签视频的音视频数据端到端地学习称为 PEMT 的多模态双向 Transformer 模型。先用 CNN 对短时（如 1-3 秒）的动态视频信息编码，再用模态专用的 Transformer（音频或视觉）建模长程依赖（如 30 秒）。接着，对各模态专用 Transformer 的输出施加多模态 Transformer，在视觉和语言领域间交换信息。不过，直接学习这样的模型会需要大量内存。为降低参数复杂度，在每个 Transformer 内跨层共享参数，参数量最多减少 80%。该 Transformer 采用基于内容感知负样本采样的对比学习训练（图 12i）。具体而言，用训练期间学得的 CNN 特征选取在视觉上与正样本相似的负样本。这项工作还比较既有方法的各种融合策略，包括早期融合（VideoBERT [17]、VL-BERT [22]）、中期融合（ViL-BERT [181]、LXMERT [21]）和晚期融合，表明中期融合是最优选择。模型在 Kinetics-700 [187] 上预训练，随后在下游视频分类任务上微调，包括 UCF101 [188] 上的短视频分类、ESC50 [189] 上的音频分类，以及 Charades [190] 和 Kinetics-Sounds [65] 上的长时动作识别。

Tan 和 Bansal [191] 引入了 voken 的概念，即与从句中提取的语言词元相关的图像。Voken（可视化词元）为语言模型提供视觉监督，使其学到更好的特征；其动机是人类通过联系视觉信息和语义概念学习语言。与其他自监督语言表征学习方法 [3], [181] 类似，他们设置 voken 预测作为辅助任务来学习表征。由于现有数据集编码的具有视觉对应关系的词元很有限，他们提出一种 voken 化方法，将语言词元映射为视觉 voken，如图 13 所示。该方法借助基于语言的检索建立映射，并将一个在小型有标签数据集（MS-COCO）上训练的模型迁移到大型数据集（Wikipedia）。

此外，建立词元–voken 映射时会考虑整个句子的上下文。用生成的词元训练所得的模型在多种 NLP 任务上表现超过了当时最优的 BERT。该模型未在视觉任务上评估，但使用视觉信息为训练语言模型提供语义对应线索，因此我们将其纳入多模态表征学习一类。

图 13 Voken 提供视觉监督 · 原文 p.16



可视化词元（voken）[191]。用紧密相关的图像对语言模型施加视觉监督，使预训练模型获得更好的特征表征。图片来自 [191]。

图中术语 | Vokens（Token-Related Images）＝词元对应的图像；Vokenization＝为语言词元建立视觉对应；Visual Supervision＝视觉监督；Visually-Supervised Language Model＝视觉监督语言模型。

视觉与语言导航（VLN）目标是根据视觉与语言输入，预测地图上的导航计划。文献 [192], [193] 已将 Transformer 用于 VLN：先对视觉–语言对进行自监督预训练，得到跨模态 Transformer，再针对特定 VLN 任务微调。这些工作学习图像区域与语言之间的注意力；Chen 等人 [194] 则提出学习语言输入与空间拓扑地图之间的跨模态注意力。拓扑地图将智能体的环境表示成图，节点为地点，边表示地点间的连通性。给定拓扑地图和自然语言输入，用 Transformer 进行 VLN 与 NLP 中的序列预测相似：在每个时间步，跨模态 Transformer 预测导航计划中拓扑地图的一个节点。语言编码与地图编码先分别由单模态编码器处理，再使用跨模态编码器（类似 LXMERT [21]）汇聚不同模态的信息。将学得的轨迹位置编码追加到地图特征上，标明地图中的位置。基于这一 Transformer 配置，文献 [194] 报告了一套能够自由探索环境、智能规划行动的完整导航系统。

CLIP [195] 是一种从文本学习图像表征的对比学习方法，目标是在一个大批次中最大化正确的图像–文本对的嵌入相似度。具体而言，给定一个含 N 对图像–文本的批次，CLIP 联合训练图像编码器和文本编码器，学习一个多模态嵌入空间，使匹配的 N 对图像–文本的余弦相似度最大化，而其余 N^2–N 对的相似度最小化。作者使用 ResNet-50 [67] 和视觉 Transformer（ViT）[132] 编码图像，并使用文献 [5] 改编的 Transformer [1] 编码文本。CLIP 在含 4 亿对图像–文本（400 million）的大型语料上训练，展现出优秀的零样本迁移能力。推理时，将类别名称输入文本编码

器，计算编码图像与全部编码文本（类别）的相似度，找出最匹配的图像–文本对。CLIP 未使用 ImageNet 训练集提供的任何监督，在 ImageNet 上仍达到惊人的 75% 零样本分类准确率。作者在 30 个不同的计算机视觉基准上进一步展示了 CLIP 的零样本迁移能力。值得注意的是，使用 ResNet 的 CLIP 在 592 块 V100 GPU 上训练了 18 天，而使用 ViT 的 CLIP 在 256 块 V100 GPU 上训练了 12 天。这突显出 CLIP 的计算开销。

3.7.2 单流 Transformer

与 ViLBERT [181]、LXMERT [21] 等双流网络不同，VisualBERT [63] 用单个 Transformer 堆栈同时建模图像与文本两个领域。将输入文本序列（如图像描述）与候选目标区域对应的视觉特征一起送入 Transformer，自动发现两领域间的关系。VisualBERT 架构与 VideoBERT [17] 有些类似（见第 3.8 节），不过它不只是聚焦烹饪视频，而是在视觉常识推理（VCR）、NLVR、VQA 和视觉指代定位等多种视觉–语言任务上评估。VisualBERT 首先以两个目标进行与任务无关的预训练（图 12-e）：其一，用图像特征和剩余文本词元预测缺失的文本词元；其二，区分给定图像的真实描述和虚假描述。作者提出，在与任务无关的预训练之后，先针对具体任务进行预训练以弥合领域差异，再最终针对下游任务微调。

Su 等人 [22] 提出多模态预训练方法，以学习可泛化到视觉常识推理和视觉问答等多模态下游任务的特征。这要求充分对齐视觉与语言线索，以学习有效的联合表征。为此，文献 [22] 在 BERT 的基础上同时输入视觉和语言特征：语言特征对应输入句中的词元；视觉特征对应输入图像感兴趣区域（RoI），由标准 Faster R-CNN 提取。模型在视觉–语言数据集 Conceptual Captions [196] 及纯语言数据集（如 Wikipedia）上预训练。损失函数与 BERT 相同：训练模型预测被掩蔽的词或视觉 RoI（图 12-f）。与 UNITER [43] 等其他工作不同，VL-BERT 声称视觉–语言匹配任务在预训练中不起作用；这一说法与后续工作 [180] 的证据不一致。在若干多模态任务上的结果表明，VL-BERT 的方法比 BERT 等仅用语言预训练的方法表现更好。

视觉与语言通用编码器（Unicoder-VL）[180] 使用大规模图像–描述对学习多模态表征。语言与图像输入送入一个 Transformer 模型（包含多个依次堆叠的编码器）以学习联合嵌入。它在预训练期间采用掩码词预测、掩码目标分类和视觉–语言匹配作为自监督任务（图 12-d）。视觉–语言匹配只在全局层面进行，即图像–句子对齐。对图文检索、零样本学习和视觉常识推理的评估表明，在这些任务上它优于 ViLBERT [181]、VisualBERT [63] 等早期模型。这显示了采用丰富的自监督任务的重要性，也支持用统一的 Transformer 架构在一个共同框架内学习多模态特征。

统一视觉–语言预训练（VLP）模型 [197] 用一个 Transformer 网络同时做编码与解码。受 BERT 启发的 VLP 模型 [17], [22], [63], [198] 则使用独立的编码器和解码器网络。将编码与解码联合建模后，统一 VLP 模型在分别针对图像描述生成和视觉问答微调时，两项任务都表现良好。共享编码与解码的动机是预训练期间需要更好地共享跨任务信息。统一模型由 12 个 Transformer 模块堆叠而成，每个模块包含一个自注意力层，其后接前馈模块。预训练的自监督目标包括掩码视觉–语言预测。作者考察了两种变体，即对掩码词进行双向预测和序列到序列预测，两种目标各用不同的上下文编码。该方法在 COCO Captions、Flick 30K Captions 和 VQA 2.0 上评估；相较早期方法，在图像描述生成及 VQA [199] 上的结果令人鼓舞。

通用图像–文本表征模型（UNITER）[43] 在四个大规模视觉–语言数据集（MS-COCO [75]、Visual Genome [200]、Conceptual Captions [196] 和 SBU Captions [201]）上预训练。所得表征可很好地迁移到 VQA、多模态检索、视觉常识推理和 NLVR 等下游任务。为着重学习视觉与语

言领域之间的关系，文献 [43] 专门设计了下列预训练任务：以另一领域的输入为条件，预测被掩蔽的视觉或文本区域；在全局（图像–文本）与局部（词–区域）两个层次对齐视觉和语言输入（图 12-a）。这些任务作为 BERT 传统掩码语言建模任务的补充，显式考虑细粒度的词–区域对齐及对 输入的条件掩蔽，这是 VL-BERT [22]、Visual-BERT [63]、Vilbert [181] 及 Unicoder-VL [180] 等早期工作未考虑的。与其他方法相同，他们采用 BERT 提出的 Transformer 架构，处理视觉和 语言嵌入。ViLBERT [181]、LXMERT [21] 等对视觉和语言输入分别使用独立的 Transformer； UNITER 则和文献 [22], [63], [180] 一样，用一个 Transformer 处理文本与图像输入。

VisualBert [63]、Uniter [43]、VL-BERT [22]、VilBERT [181] 和 Unicoder-VL [180] 的 VLP 模型都拼接图像与文本特征，由自注意力自动发现跨模态关系。这可能使图像中语义概念的视觉指代定位更困难。为解决此问题，目标–语义对齐预训练（Oscar）[44] 先用目标检测器获取目标标签，然后以这些标签为媒介将相关视觉特征与语义信息对齐（图 12-b）。其动机是：文本内容通常与图中主要目标相关，因此将相应图像标签显式加入输入，可更好地关注视觉特征。像 BERT [3] 一样，Oscar 用掩码词元损失做 VLP：随机掩蔽文本输入和图像标签中的不同词元，再用模型预测缺失的词元。它还使用对比损失，区分真实的图像–标签对与带噪或虚假的配对。在 VQA、跨模态检索、自然语言推理及图像描述生成任务上微调后，所得表征优于不使用目标标签的 VLP 方法。近期的 VinVL [202] 将 Oscar 扩展到目标检测任务，用调整后的预训练方案学习以目标实例为中心的视觉–语言关系。该模型在 MS-COCO、OpenImages、Visual Genome、Objects365 的数据集组合上训练；实验表明，它能准确地把语义属性和视觉信息关联起来，并更好地迁移到下游视觉理解任务。

3.7.3 用于视觉指代定位的 Transformer

调制 DETR（MDETR）[203] 分别用 CNN 和 BERT 作为骨干网络，从图像与文本输入提取特征。将视觉和文本特征分别线性投影到一个共享空间，拼接后送入与 DETR 类似的 Transformer 模型，预测定位文本中查询所对应的目标边界框。训练所用的损失使模型针对每个预测框，在与之相关的所有文本查询词元上预测均匀分布。额外的对比损失项保证视觉嵌入和文本嵌入间的对应。TransVG [204] 是一种简洁设计：在 Transformer 模块中融合视觉和文本特征，使用随这些特征一同输入该模块的可学习词元，直接回归查询对应的边界框。Referring Transformer [205] 也是简洁的单阶段设计；它在 Transformer 编码器中融合文本与图像特征，再通过 Transformer 解码器直接回归边界框或分割掩码。Visual Grounding with Transformer [206] 采用编码器–解码器架构；编码器的两个不同分支并行处理视觉词元（从预训练 CNN 提取的特征）和文本词元（经 RNN 模块解析的特征），通过跨模态注意力得到文本引导的视觉特征。解码器随后计算文本查询和视觉特征之间的注意力，再预测针对各查询的边界框。

3.8 视频理解

现有音视频数据分析方法一般从较短的视频（几秒以内）学习表征，因此只能编码短程依赖 [1], [32]。活动识别 [71], [187], [207]–[209] 等单模态或多模态学习任务都需要建模长程依赖。下文介绍近期借助 Transformer 网络的表达能力应对这一挑战的方法。值得注意的是，其中若干工作 [17], [18], [182], [210] 仍用（预训练）CNN 对视频中的图像级或帧级特征编码，再使用 Transformer 对更广的上下文建模。文献 [209], [211]–[213] 是少数例外，它们也使用基于 ViT 的骨干网络提取帧级特征。

3.8.1 视频与语言联合建模

VideoBERT [17] 借助 Transformer 网络及自监督学习的优势，学习有效的多模态表征。具体而言，它把预测被掩蔽的视觉和语言词元作为代理任务（图 12-c），由此建模对视频理解很重要的高层语义和长程时间依赖。给定视频，文献 [17] 用现成的语音识别系统将语音转为文本，并使用向量量化（聚类）从预训练的视频分类模型中获得视觉特征。随后，直接将 BERT 用在拼接后的语言与视觉词元序列上，学习两者的联合分布。模型可以用纯文本、纯视频以及视频加文本三类数据训练。所得模型展示了颇有趣的跨模态预测能力，例如给定文本（如说明文字或菜谱）生成视频，以及基于视频预测未来。视频加文本模型用视觉–语言对齐任务学习跨模态关系。该代理任务定义很简单：给定 [cls] 词元的潜在状态，预测句子在时间上是否与视觉词元序列对齐。此外，所得表征被证明对动作分类、零样本分类及视频描述生成等下游任务很有用。

Zhou 等人 [210] 探索使用掩码 Transformer 进行密集视频描述生成。这项任务需要为视频中发生的所有事件生成语言描述。已有方法一般顺序执行：先检测事件，再在独立模块中生成描述。文献 [210] 提出统一的 Transformer 网络联合处理两项任务，将事件检测与描述生成整合在一起。首先使用视频编码器得到逐帧表征，再用两个解码模块分别提出视频事件和对应的文字描述。由于处理的是未经剪辑的视频，描述解码器使用掩码网络，专注于描述单个候选事件。值得注意的是，文献 [210] 首次尝试用非循环模型进行密集视频描述生成，并在编码器中对 CNN 提取的特征施加自注意力，建模视频帧间宽范围的上下文。在 ActivityNet Captions [214] 和 YouCookII [215] 上的实验显示，相比此前的循环网络和两阶段方法，结果有良好提升。

3.8.2 视频动作识别

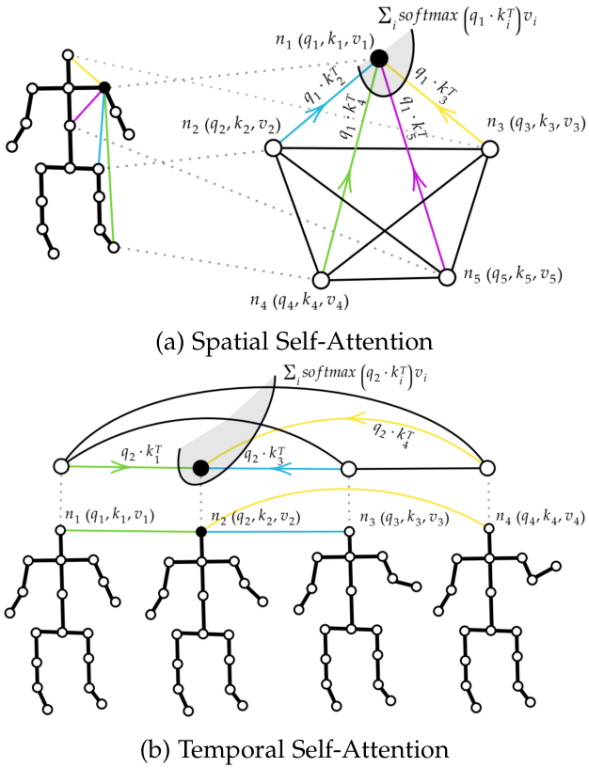
传统的基于 CNN 的视频分类方法，一般在有限的时间区间内进行三维时空处理以理解视频。Neimark 等人 [211] 提出视频 Transformer 网络（VTN），先用二维 CNN 获取逐帧特征，再用 Transformer 编码器（Longformer [103]）学习时间关系。Longformer 的复杂度为 O(n)，因而适于处理任意长度 n 的长序列。将类别词元送入全连接层，识别动作或事件。在空间特征上使用 Transformer 编码器有两项优势：(a) 单次前向处理就能处理完整视频；(b) 避免了昂贵的三维卷积，因而显著提高训练与推理效率。这使得 VTN 尤其适于建模长视频，因为其中实体间的相互作用可能分布在整段视频中。作者在 Kinetics-400 [71] 上使用多种骨干网络（ResNet [67]、ViT [11] 和 DeiT [12]）开展实验，取得了有竞争力的性能。

Girdhar 等人 [18] 使用一种 Transformer 架构变体，汇聚视频中与特定人物相关的上下文线索，用于动作分类和定位。模型先采用 Faster-RCNN [125] 式的流程：骨干网络生成特征，送入区域提议网络得到候选目标；然后通过 RoI 池化生成各目标的特征。接下来，在目标特征上级联多个多头自注意力层 [1]。在每个 Transformer 单元内，特定人物的特征作为查询（Q），相邻视频片段的特征作为键（K）和值（V）。从中提取 K、V 和 Q 的输入特征图中显式编码了位置信息，从而把位置信息纳入自注意力。对于给定的 $400 \times 400 \times 64$ 视频片段，键和值张量的形状是 $16 \times 25 \times 25 \times 128$，而查询是 128 维向量。尽管文献 [18] 只使用 RGB 流，如果像竞争方法那样增加光流、音频信号等模态，计算复杂度会进一步增加。此外，研究发现该 Transformer 模型在动作定位上的表现并不理想，可能因为它倾向于纳入全局信息。所以，对动作定位和分割等要求精确划定边界的问题，全局与局部上下文之间需要适当权衡。

基于骨架表征的人体动作识别要求理解给定帧中各身体关节之间的关系，也要理解视频不同帧之间的关系。Plizzari 等人 [216] 提出双流 Transformer 网络建模这些关系。他们引入空间自注

意力（SSA）建模不同关节间的关系（图 14a），并用时间自注意力（TSA）捕获长程帧间依赖（图 14b）。他们先用小型残差网络从骨架数据中提取特征，再用 SSA、TSA 模块处理这些特征图。SSA 独立找出每对关节的相关性，TSA 则专注于某一关节的特征在时间维度上如何随帧变化。SSA 的目的是找出周围关节之间的关系，类似于 Transformer 联系短语中的不同词；TSA 则寻找各帧之间的长程关系，类似 NLP 中对短语之间建模。双流模型在 NTU-RGB+D 60 [217] 和 NTU-RGB+D 120 [218] 上取得了当时最优结果。

图 14 骨架数据的空间和时间自注意力 · 原文 p.18



骨架数据表征的空间与时间注意力。用两个专门的自注意力模块，分别建模身体关节之间的关系和帧间依赖。图片来自 [216]。

图中术语 | (a) Spatial Self-Attention＝空间自注意力：同帧跨关节；(b) Temporal Self-Attention＝时间自注意力：同一关节跨帧。

多尺度视觉 Transformer（MViT）[219] 逐步增加通道容量并降低视频的时空分辨率，从而构建特征层级。他们引入多头池化注意力，在金字塔结构中逐步改变视觉分辨率。TimeSFormer [213] 将 ViT [132] 扩展到视频，将其视为从各帧提取的图像块序列。为捕捉时空关系，他们提出分离注意力，即在每个模块中将空间注意力与时间注意力分开施加。TimeSFormer 在动作识别上达到当时最优水平，也可用于时长超过一分钟的片段。另一个值得注意的纯 Transformer 模型是视频视觉 Transformer（ViViT）[212]。它先提取时空词元，再用高效的分解式自注意力编码词元间的关系。但这类模型需要使用在图像上预训练的模型初始化，才能有效训练 ViT。同期也有工作用 ViT 与自监督学习训练可靠的预训练模型。其中一项重要工作是长短对比学习（LSTCL）框架 [220]：它把从不同时间尺度（短时与长时）重建表征作为辅助学习任务，在下游任务上表现良好。

3.8.3 视频实例分割

视频实例分割 Transformer（VisTR）[209] 将 DETR [13] 扩展到视频目标实例分割（VIS）任务。它通过 CNN 骨干网络从一组视频帧提取局部特征，并像 DETR 一样使用 Transformer 编码器与解码器，将实例分割转化为序列到序列预测。先把输入的帧级特征拼接成片段表征，Transformer 再以各帧间一致的顺序输出实例预测。这样就把目标检测和跟踪整合到一个统一架构中。预测结果通过二分匹配与真值配对。与 Mask R-CNN [127] 类似，另设一个使用自注意力和三维卷积的头部，预测实例掩码。总体而言，该方法在 YouTube VIS 数据集 [221] 的单模型方法中具有竞争力，但略逊于 MaskProp [222] 等更复杂的 CNN 模型。

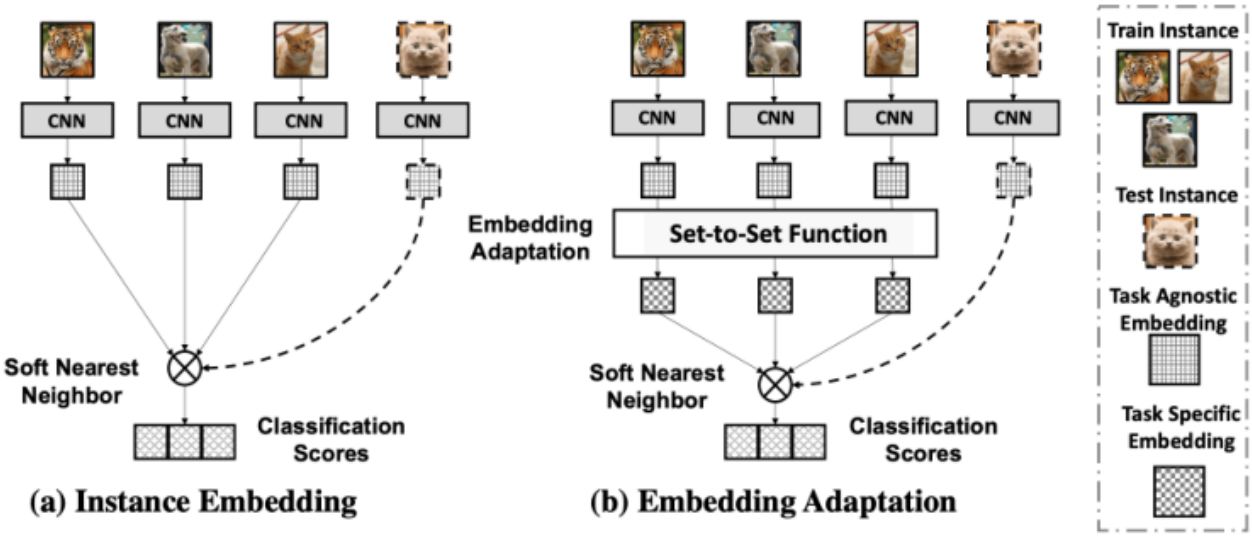
3.9 小样本学习中的 Transformer

在小样本学习中，推理时提供一个支持集，以适配一组新类别。已有方法使用 Transformer 在支持集上学习集合到集合的映射 [26]，或学习给定查询输入与支持集样本的空间关系 [25]。就绝对性能而言，查询图像与支持集图像间基于图像块的空间自注意力，优于文献 [26] 学习的图像级关联。不过，逐块计算注意力的代价较高。下文详细说明这些方法。

Doersch 等人 [25] 探索在小样本细粒度分类中使用自监督和 Transformer，此时训练与评估阶段的数据分布不匹配。他们开发了 Cross-Transformer，将查询图像与支持集中的少量样本联系起来。具体来说，Transformer 找出查询图像与支持集图像中空间上相似的区域，再用对应特征为查询作出分类。Transformer 中的查询来自查询图像的网格特征；类似地，用支持图像的网格特征构造键和值，再由此得到经过注意力加权的输出。该方法配合基于对比自监督的训练机制，在难度较高的 Meta-dataset [223] 上达到最佳性能。

Ye 等人 [26] 提出在推理时用 Transformer 模块，将基础类别上学到的小样本嵌入适配到目标小样本类别。这样得到特定于任务的嵌入，在小样本分类等判别任务上效果更好。作者还评估了多种其他集合到集合的函数，例如图卷积网络 [224]、双向 LSTM [32] 和 DeepSets [225]；基于 Transformer 的映射表现最佳。这归因于 Transformer 擅长利用上下文，能在任务上插值与外推，并具备排列不变性，同时维持相对较低的参数复杂度。文献 [26] 的 Transformer 架构遵循标准模型 [1]，用对比损失函数来适配嵌入，以保留判别能力（图 15）。所得模型在归纳式、直推式和广义小样本学习任务中都表现强劲。

图 15 FEAT 的实例嵌入与任务适配 · 原文 p.19



FEAT [26] 概览。传统小样本学习（FSL）的实例嵌入方法在所有任务上使用同一嵌入函数 (a)，而 FEAT 使用集合到集合的函数调整嵌入，使其适应各个小样本任务 (b)。作者评估了几种集合到集合的函数，发现 Transformer 模块最适合小样本学习。图片来自 [26]。

图中术语｜Instance Embedding＝实例嵌入；Embedding Adaptation＝嵌入适配；Set-to-Set Function＝集合到集合函数；Soft Nearest Neighbor＝软最近邻；Task-Agnostic / Task-Specific Embedding＝任务无关/特定嵌入。

Liu 等人 [226] 学习了一个基于多头自注意力的模块，整合由 Meta-dataset [223] 不同领域的数据训练的各模型所学到的视觉表征。通用表征 Transformer（URT）层动态地调整不同领域专用骨干网络所输出表征的权重，结果表明它能有效处理多种数据分布下的小样本任务。

3.10 用于聚类的 Transformer

聚类通过将相似数据点归为一组来发现数据中的结构，广泛用于数据可视化与解读、异常检测和开放集分类。虽然已经为集合预测问题开发了神经网络 [225], [227]，但这些方法逐点处理集合元素，可能丢失点间关系的信息。近期工作采用直接处理集合输入的集合 Transformer（ST）[228] 进行摊销聚类。摊销聚类是一项有挑战的任务：学习一个参数化函数，将输入点集映射到相应的聚类中心。Lee 等人 [228] 提出用包含多头自注意力模块 [1] 的 Transformer 学习这种映射。这样的 Transformer 在设计上具有排列不变性，能同时编码输入点之间的成对关系与高阶关系。然而，完整 Transformer 每个自注意力层的计算成本为 $O(n^2)$，其中 n 为集合中的点数。ST 通过诱导自注意力模块将其降至 $O(mn)$：该模块使用低秩投影（$H \in \mathbb{R}^m$），从而能处理大集合。模型以最大化高斯混合分布（MoG）的似然为目标训练，学习最佳参数。因此，给定一组数据点，ST 可估计 MoG 参数。除了摊销聚类，ST 还是一种通用框架，可解决其他集合输入问题，如统计输入集合中的不同元素数、多实例学习、集合异常检测和三维点云分类。较近期的文献 [229] 采用顺序生成聚类的方式改进文献 [228]，因而可以把点分配到数量可变的聚类。

3.11 用于三维分析的 Transformer

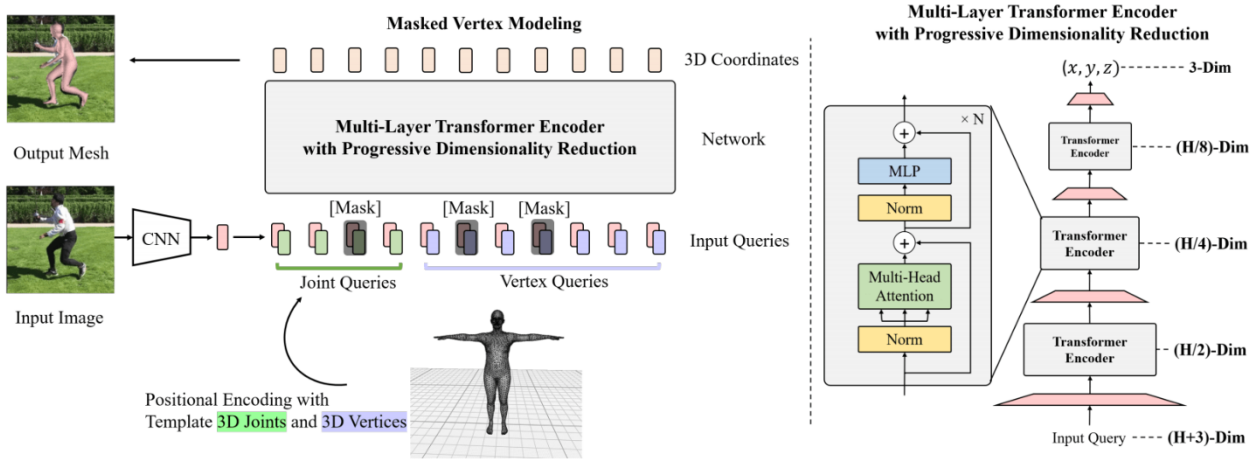
三维点云表征具有不规则性（点数可变）与排列不变性，Transformer 为编码三维数据点间丰富的关系提供了一种很有前景的机制。近期工作 [230], [231] 正是利用 Transformer 学习集合函数的能力。具体而言，文献 [230] 引入 Point Transformer，通过向量注意力学习各通道的权重；文献 [231] 则提出显式编码本地三维结构的另一种设计。文献 [45] 利用 Transformer 的非局部性，开发准确的人体姿态与网格重建算法。下文讨论这几种方法。

作为集合算子，自注意力适合处理点云：这种三维数据表征要求对点的数量和排列保持不变。Zhao 等人 [230] 提出 Point Transformer 层，在三维点的局部邻域内应用自注意力。该层以向量化的自注意力网络（SAN）[82] 为基础，使用向量表示注意力权重。此外，为表示位置信息，注意力向量和变换后的特征（值向量）都加入位置编码。将 Point Transformer 层夹在两个线性层之间构成 Point Transformer 模块，再在网络架构中多次堆叠。该设计还包括用于减少或增加输入点数的下采样/上采样过渡模块，采用典型的编码器–解码器式流程。所得架构在三维分类和分割任务中展现出良好前景。

点云 Transformer（PCT）[231] 与文献 [230] 是同期工作，其设计动机也是 Transformer 的排列不变性。但相较于文献 [230]，它更直接地基于传统 Transformer 架构 [1]，不使用向量注意力。主要修改包括基于三维坐标的位置编码、偏移注意力模块及对点云局部三维结构编码的邻域嵌入。具体而言，偏移注意力层通过逐元素相减，计算自注意力加权后的特征与输入特征的差。局部邻域嵌入求的是点组之间的自注意力关系，而非单独三维点之间的关系。显式引入局部邻域信息使该架构比文献 [230] 更高效。该方法在 ModelNet40 [232] 和 ShapeNet [233] 上的三维形状分类、法向估计与分割任务中表现出良好前景。

网格 Transformer（METRO）[45] 面向单张二维图像的三维人体姿态和网格重建。其中一项关键挑战是准确学习身体关节与网格顶点之间的非局部相互作用，例如手与脚之间的关系。Transformer 的表达能力用于同时建模网格顶点间的关系，以及顶点与身体关节的关系。自注意力可关注网格顶点的任意组合，从而编码非局部关系。多层 Transformer 按顺序降维，将二维图像映射到三维网格。使用每个顶点和身体关节的三维坐标 (x,y,z) 作位置编码。类似于 NLP 的掩码语言建模，METRO 采用掩码顶点建模（MVM），随机掩蔽一定比例的输入查询（见图 16）。Transformer 的任务是回归所有关节和顶点，这有助于编码两者间的相互依赖。METRO 在 Human3.6M [234] 和 3DPW [235] 的人体网格重建任务中达到当时最优结果。由于不依赖参数化网格模型，该方法能很好地泛化到三维手部重建等其他重建任务 [236]。总体来看，这是将 Transformer 用于三维人体重建的首次尝试，取得了相当好的结果。

图 16 METRO 的人体网格重建 · 原文 p.23



Mesh Transformer 架构。关节和顶点查询添加位置嵌入后，通过多层自注意力，联合回归关节与网格顶点的三维坐标。图片来自 [45]。

图中术语 | Masked Vertex Modeling＝掩码顶点建模；Joint/Vertex Queries＝关节/顶点查询；Multi-Layer Transformer Encoder with Progressive Dimensionality Reduction＝渐进降维的多层 Transformer 编码器；Output Mesh＝输出网格。

4 开放挑战与未来方向

尽管 Transformer 模型性能优异且有一些有趣的显著特征（表 1），在实际应用中仍面临若干挑战（表 2）。其中最重要的瓶颈是需要大量训练数据，并伴有高昂的计算成本。Transformer 模型的可视化和解释也有一定困难。本节概述这些挑战，介绍近期应对这些局限的若干工作，并指出仍待研究的问题。

表 1 不同视觉任务中的 Transformer 设计 · 原文 p.21

具有代表性的计算机视觉应用中，各种 Transformer 变体采用的关键设计选择。主要差别在于损失函数的选择、架构改动、位置嵌入方式以及输入数据模态。

任务	方法	设计亮点（侧重与标准形式的差别）	输入数据类型	标签类型	损失
图像分类	ViT [11]	直接采用 NLP Transformer 编码器处理图像；线性嵌入图像块并加入适用于编码器的位置嵌入。	2D 图像	类别标签	交叉熵
图像分类	DeiT [12]	Transformer 为学生、CNN 为教师；用蒸馏词元估计教师给出的标签；在类别词元和蒸馏词元之间计算注意力。	2D 图像	类别标签	交叉熵；基于 KL 散度的蒸馏损失
图像分类	CLIP [195]	在图像–文本对上联合训练图像与文本编码器，使匹配对的相似度最大，不匹配对的相似度最小。	2D 图像和文本	图像–文本对	对称交叉熵

任务	方法	设计亮点（侧重与标准形式的差别）	输入数据类型	标签类型	损失
目标检测	DETR [13]	以线性投影层降低 CNN 特征维度；在编码器和解码器的各多头自注意力层加入空间位置嵌入；在解码器各多头自注意力层再加入目标查询（输出位置编码）。	2D 图像	类别标签	基于预测与真值之间二分匹配的匈牙利损失
目标检测	D-DETR [14]	可变形 Transformer 由可变形注意力层构成，为 Transformer 引入稀疏先验；采用多尺度注意力模块。	2D 图像	类别标签	匈牙利损失
小样本学习	CT [25]	自监督预训练；使用与查询对齐的类别原型，在支持集图像和查询图像之间建立空间对应。	2D 图像	预训练无标签，小样本学习使用类别标签	归一化交叉熵
图像上色	ColTran [24]	条件行/列多头注意力层；渐进多尺度上色方案。	2D 图像	2D 图像	图像的负对数似然
动作识别	ST-TR [216]	空间和时间自注意力，用于处理骨架关节等图结构数据。	骨架	动作类别	交叉熵
超分辨率	TTSR [16]	用 Transformer 模块增强纹理；用相关性嵌入计算低分辨率图像与参考图像的相关性。	2D 图像	2D 图像	重建损失；定义在预训练 VGG19 特征上的感知损失
多模态学习	Oscar [44]	Transformer 层联合处理图文三元组 [词、标签、特征]；用掩码词元表征文本数据。	2D 图像	描述、类别标签、目标标签	被掩蔽词元的负对数似然；对比二元交叉熵
3D 分类/分割	PT [230]	Point Transformer 模块；用下采样过渡模块减少点集的点数；用上采样过渡模块做稠密预测。	CAD 模型、3D 目标部件分割	目标和形状类别	交叉熵
3D 网格重建	METRO [45]	在各 Transformer 层中逐步降维；以 3D 关节和 3D 顶点坐标作位置编码；掩码顶点/关节建模。	2D 图像	3D 网格 + 人体姿态	在 3D 空间和 2D 投影中对网格顶点和关节计算 L1 损失
视觉与语言导航	Chen 等人 [194]	对语言和地图输入分别用单模态编码器，再用跨模态 Transformer；在地图编码器中加入轨迹位置编码。	指令文本 + RGBD 全景图 + 环境拓扑地图	导航计划	对节点与 [stop] 动作计算交叉熵
指代表达图像分割	CMSA [15]	多模态特征；在多个层级进行跨模态自注意力，并用学到的门控进行融合。	2D 图像 + 语言表达	分割掩码	二元交叉熵损失
视频分类	Lee 等人 [182]	处理实值音视频信号而非词元；用对比学习做预训练；端到端学习多模态 Transformer。	音视频	活动标签	对比 InfoNCE 损失与二元交叉熵

对照任务、方法、输入、监督标签和损失。

表 1 英文原表 · p.21

Task	Method	Design Highlights (focus on differences with the standard form)	Input Data Type	Label Type	Loss
Image Classification	ViT [11]	Directly adopted NLP Transformer Encoder for images, Mechanism to linearly embed image patches with positional embedding suitable for the Encoder.	2D Image	Class labels	Cross-entropy
	DeiT [12]	Transformer as s student while CNN as a teacher, Distillation tokens to produce estimated labels from teacher, Attention between class and distillation tokens.	2D Image	Class labels	Cross-entropy, Distillation loss based on KL-divergence
	CLIP [195]	Jointly train image and text encoders on image-text pairs, to maximize similarity of valid pairs and minimize otherwise	2D Images & texts	Image-text pairs	Symmetric cross-entropy
Object Detection	DETR [13]	Linear projection layer to reduce CNN feature dimension, Spatial positional embedding added to each multi-head self-attention layer of both encoder and decoder. Object queries (output positional encoding) added to each multi-head self-attention layer of decoder.	2D Image	Class labels	Hungarian loss based on bipartite matching between predicted and ground truths
	D-DETR [14]	Deformable Transformer consists of deformable attention layers to introduce sparse priors in Transformers, Multi-scale attention module.	2D Image	Class labels	Hungarian loss
Low Shot Learning	CT [25]	Self-supervised pretraining, Query-aligned class prototypes that provide spatial correspondence between the support-set images and query image.	2D Image	Pretraining without labels and few-shot learning with Class labels	Normalized Cross-entropy
Image Colorization	ColTran [24]	Conditional Row/column multi-head attention layers, Progressive multi-scale colorization scheme.	2D Image	2D Image	Negative log-likelihood of the images
Action Recognition	ST-TR [216]	Spatial and Temporal self-attention to operates on graph data such as joints in skeletons.	Skeleton	Action Classes	Cross-entropy
Super-resolution	TTSR [16]	Texture enhancing Transformer module, Relevance embeddings to compute the relevance between the low-resolution and reference image.	2D Image	2D Image	Reconstruction loss, Perceptual loss defined on pretrained VGG19 features.
Multi-Model Learning	Oscar [44]	Transformer layer to jointly process triplet representation of image-text [words, tags, features], Masked tokens to represent text data.	2D Image	Captions, Class labels, Object tags	Negative log-likelihood of masked tokens, Contrastive binary cross-entropy
3D Classification/Segmentation	PT [230]	Point Transformer block, Transition down block to reduce cardinality of the point set, Transition up for dense prediction tasks.	CAD models, 3D object part segmentation	Object and shape categories	Cross-entropy
3D Mesh Reconstruction	METRO [45]	Progressive dimensionality reduction across Transformer layers, Positional Encoding with 3D joint and 3D vertex coordinates, Masked vertex/joint modeling.	2D Image	3D Mesh + Human Pose	L_1 loss on mesh vertices and joints in 3D and 2D projection.
Vision and Language Navigation	Chen *et al.* [194]	Uni-modal encoders on language and map inputs followed by a cross-modal transformer, Trajectory position encodings in the map encoder.	Instruction text + RGBD panorama + Topological Environment Map	Navigation Plan	Cross-entropy over nodes and `[stop]` action
Referring Image Segmentation	CMSA [15]	Multimodal feature, Cross-modal self-attention on multiple levels and their fusion using learned gates.	2D Image + Language expression	Segmentation mask	Binary cross-entropy loss
Video Classification	Lee *et al.* [182]	Operates on real-valued audio-visual signals instead of tokens, Contrastive learning for pre-training, End-to-end multi-modal transformer learning.	Audio-Visual	Activity labels	Contrastive InfoNCE loss and Binary cross-entropy

表 2 不同任务的方法、指标与局限 · 原文 p.22

采用 Transformer 的不同方法在各项任务上的优势和局限。CT：Cross Transformers；AP：平均精度；mAP：平均 AP；IoU：交并比；FID：Fréchet Inception 距离；MPJPE：平均每关节位置误差；MPVE：平均每顶点误差。

任务 / 方法	指标	数据集 / 性能	优势	局限
图像分类 ViT [11] ICLR'21	Top-1 准确率	ImageNet 88.55	a) 首次将 Transformer（全局自注意力）直接用于图像块；b) 架构不含卷积；c) 优于 ResNet 等 CNN 模型。	a) 需要在大规模数据上训练，例如 3 亿（300 million）张图像；b) 迁移到新任务时需谨慎；c) 需使用含 6.32 亿（632 million）参数的大模型才能达到当时最优。
图像分类 DeiT [12] arXiv'20	Top-1 准确率	ImageNet 83.10	a) 仅用 ImageNet 成功训练 Transformer；b) 引入基于注意力的蒸馏；c) 用小型（8600 万，即 86 million 参数）Transformer 取得有竞争力的结果。	a) 需使用预训练 CNN 教师模型，性能因此取决于教师模型的质量。
图像分类 Swin-T [36] arXiv'21	Top-1 准确率	ImageNet 84.5	a) 为分类、检测、分割等不同视觉任务提供通用骨干网络；b) 采用移动窗口运算的层级式设计。	a) 在较小的数据集上难以从头训练；b) 自注意力运算具有固有的二次计算复杂度。
小样本学习 CT [25] NeurIPS'20	Top-1 准确率	ImageNet 62.25 COCO 60.35	a) 自监督预训练不需要人工标签；b) 使用 Transformer 做动态推理，达到当时最优。	该方法在缺少纹理等空间细节的数据集上的表现有限。
目标检测 DETR [13] ECCV'20	AP	COCO 44.9	a) 使用 Transformer 实现目标检测的端到端训练；b) 不再需要手工设计的后处理。	a) 小目标表现较差；b) 需要较长的训练时间才能收敛。
目标检测 D-DETR [14] ICLR'21	AP	COCO 43.8	a) 小目标表现优于 DETR [13]；b) 比 DETR [13] 收敛更快。	获得当时最优的 52.3 AP，但需使用两阶段检测器设计及测试时增强。
图像上色 ColTran [24] ICLR'21	FID	ImageNet 19.71	a) 首次成功将 Transformer 用于图像上色；b) FID 达到当时最优。	a) 未进行端到端训练；b) 仅适用于 256×256 图像。
动作识别 ST-TR [216] arXiv'20	Top-1 准确率	NTU 60/120 94.0/84.7	a) 将 Transformer 成功用于对空间和时间上的人体关节关系建模；b) 达到当时最优。	Transformer 不直接处理关节，而处理 CNN 提取的特征，所以整体模型依赖手工架构设计。

任务 / 方法	指标	数据集 / 性能	优势	局限
超分辨率 TTSR [16] CVPR'20	PSNR / SSIM	CUFED5 27.1 / 0.8 Sun80 30.0 / 0.81 Urban100 25.9 / 0.78 Manga109 30.1 / 0.91	a) 用注意力达到当时最优的超分辨率性能；b) 受 Transformer 启发的新架构能处理多尺度特征。	a) Transformer 不直接处理图像，而处理卷积网络提取的特征；b) 可训练参数量大；c) 计算开销高。
多模态学习 ViLBERT [181] NeurIPS'19	准确率 / mAP (R@1)	VQA [183] / 检索 [239] 70.6/58.2	a) Transformer 架构能够结合文本与视觉信息，理解任务间依赖；b) 能在无标签数据集上预训练。	a) 预训练需要大量数据；b) 需要在新任务上微调。
多模态学习 Oscar [44] ECCV'20	准确率 / mAP (R@1)	VQA [240] / COCO 80.37/57.5	a) 利用目标标签提供新的监督信号，以对齐文本与图像；b) 达到当时最优。	需要通过预训练目标检测器提供额外的监督，因此性能取决于检测器的质量。
多模态学习 UNITER [43] ECCV'20	准确率 / 平均值 (R@1/5/10)	VQA [183] / Flickr30K [241] 72.47/83.72	学习文本与图像间的细粒度关系对齐。	训练 Transformer 需要大型多任务数据集，导致计算成本高。
3D 分析 Point Transformer [230] arXiv'20	Top-1 准确率 IoU	ModelNet40 [232] 92.8；85.9	a) 基于 Transformer 的注意力可以处理无序、非结构化的点集；b) 架构具有排列不变性。	a) 相较此前最优仅有适度提升；b) 可训练参数量大，约为 PointNet++ [242] 的 6 倍。
3D 分析 METRO [45] arXiv'20	MPJPE PA-MPJPE MPVE	3DPW [235] 77.1；47.9；88.2	a) 不依赖参数化网格模型，故易于扩展到不同物体；b) 使用 Transformer 达到当时最优。	依赖手工设计的网络架构。

同一任务在不同实验配置下的结果不能直接视为完全可比。并列数据集与并列数值保持原表顺序。

表 2 英文原表 · p.22

Task	Method	Metric	Dataset	Performance	Highlights	Limitations
Image Classification	ViT [11] ICLR'21	Top-1 Acc.	ImageNet	88.55	**a)** First application of Transformer (global self-attention) directly on image patches, **b)** Convolution-free network architecture, **c)** Outperforms CNN models such as ResNet.	**a)** Requires training on large-scale data *e.g.*, 300-Million images, **b)** Requires careful transfer learning to the new task, **c)** Requires large model with 632-Million parameters to achieve SOTA results.
	DeiT [12] arXiv'20	Top-1 Acc.	ImageNet	83.10	**a)** Successfully trains Transformer on ImageNet only, **b)** Introduces attention-based distillation method. **c)** Produces competitive performance with small (86-Million parameters) Transformers.	**a)** Requires access to pretrained CNN based teacher model thus performance depends on the quality of the teacher model.
	Swin-T [36] arXiv'21	Top-1 Acc.	ImageNet	84.5	**a)** Provides a general purpose backbone for different vision tasks e.g., classification, detection and segmentation **b)** A hierarchical design using shifted-windows operation.	**a)** Hard to train from scratch on smaller datasets **b)** Quadratic compute complexity inherent to the self-attention operation.
Low-Shot Learning	CT [25] NeurIPS'20	Top-1 Acc.	ImageNet COCO	62.25 60.35	**a)** Self-supervised pre-training mechanism that does not need manual labels, **b)** Dynamic inference using Transformer achieving stat-of-the-art results.	Proposed algorithm is limited in its capacity to perform on datasets that lack spatial details such as texture.
Object Detection	DETR [13] ECCV'20	AP	COCO	44.9	**a)** Use of Transformer allows end-to-end training pipeline for object detection, **b)** Removes the need for hand-crafted post-processing steps.	**a)** Performs poorly on small objects, **b)** Requires long training time to converge.
	D-DETR [14] ICLR'21	AP	COCO	43.8	**a)** Achieves better performance on small objects than DETR [13], **b)** Faster convergence than DETR [13]	Obtain SOTA results with 52.3 AP but with two stage detector design and test time augmentations.
Image Colorization	ColTran [24] ICLR'21	FID	ImageNet	19.71	**a)** First successful application of Transformer to image colorization, **b)** Achieves SOTA FID score.	**a)** Lacks end-to-end training, **b)** limited to images of size 256×256.
Action Recognition	ST-TR [216] arXiv'20	Top-1 Acc.	NTU 60/120	94.0/84.7	**a)** Successfully applies Transformer to model relations between body joints both in spatial and temporal domain, **b)** Achieves SOTA results.	Proposed Transformers do not process joints directly rather operate on features extracted by a CNN, thus the overall model is based on hand-crafted design.
Super-Resolution	TTSR [16] CVPR'20	PSNR/ SSIM	CUFED5 Sun80 Urban100 Manga109	27.1 / 0.8 30.0 / 0.81 25.9 / 0.78 30.1 / 0.91	**a)** Achieves state-of-the-art super-resolution by using attention, **b)** Novel Transformer inspired architectures that can process multi-scale features.	**a)** Proposed Transformer does not process images directly but features extracted by a convolution based network, **b)** Model with large number of trainable parameters, and **c)** Compute intensive.
Multi-Model Learning	ViLBERT [181] NeurIPS'19	Acc./ mAP (R@1)	VQA [183]/ Retrieval [239]	70.6/ 58.2	**a)** Proposed Transformer architecture can combine text and visual information to understand inter-task dependencies, **b)** Achieves pre-training on unlabelled dataset.	**a)** Requires large amount of data for pre-training, **b)** Requires fine tuning to the new task.
	Oscar [44] ECCV'20	Acc./ mAP (R@1)	VQA [240]/ COCO	80.37/57.5	**a)** Exploit novel supervisory signal via object tags to achieve text and image alignment, **b)** Achieves state-of-the-art results.	Requires extra supervision through pre-trained object detectors thus performance is dependent on the quality of object detectors.
	UNITER [43] ECCV'20	Acc./ Avg. (R@1/5/10)	VQA [183]/ Flickr30K [241]	72.47/83.72	Learns fine-grained relation alignment between text and images	Requires large multi-task datasets for Transformer training which lead to high computational cost.
3D Analysis	Point Transformer [230] arXiv'20	Top-1 Acc. IoU	ModelNet40 [232]	92.8 85.9	**a)** Transformer based attention capable to process unordered and unstructured point sets, **b)** Permutation invariant architecture.	**a)** Only moderate improvements over previous SOTA, **b)** Large number of trainable parameters around 6× higher than PointNet++ [242].
	METRO [45] arXiv'20	MPJPE PA-MPJPE MPVE	3DPW [235]	77.1 47.9 88.2	**a)** Does not depend on parametric mesh models so easily extendable to different objects, **b)** Achieves SOTA results using Transformers.	Dependent on hand-crafted network design.

4.1 高计算成本

如第 1 节所述，Transformer 的一项优势是能灵活扩展到参数很多的模型。这项特性能让我们训练超大模型，但也导致较高的训练和推理成本（表 3 详细比较 CNN 与 ViT）。例如，含 1.09 亿个参数（109 million）的 BERT 基础模型 [3] 训练约消耗 1.89 peta-flop days[2]；拥有 1750 亿个参数（175 billion）的近期 GPT3 模型 [6] 则约需 3640 peta-flop days 才能训练完成，达到惊人的约 1925 倍。这也意味着高昂的价格：一项估算 [237] 认为，OpenAI 训练 GPT3 可能花费 460 万美元（4.6 million USD）。此外，这些大规模模型需要大幅压缩（如通过蒸馏），才能在现实环境中实用。

原文脚注 2. peta-flop day（千万亿次浮点运算/秒 • 日）是计算量单位，等于以每秒 10^{15} 次神经网络运算的速度连续运算一整天。

表 3 视觉 Transformer 与 CNN 的参数量和性能 · 原文 p.23

比较视觉 Transformer 和 CNN 模型的参数复杂度，以及在 ImageNet 验证集上的 Top-1 准确率（%）。为进行直接比较，所选模型均在 ImageNet 上从头训练，输入大小均为 224×224。⋆ 表示纯 CNN 方法。

方法	参数量（百万）	GFLOPs	Top-1 准确率（%）
ResNet18 [67]⋆	11.7	1.8	69.8
EfficientNet-B3 [87]⋆	12.0	1.8	81.6
DeiT-T [12]	5.7	1.3	72.2
T2T-ViTt-7 [35]	5.0	1.3	71.7
LocalViT-T [107]	5.9	1.3	74.8
CrossViT-T [104]	6.9	1.6	73.4
PVTv1-T [93]	13.2	1.9	75.1
ResT-Lite [110]	10.5	1.4	77.2
CaiT-XXX-24 [243]	12.0	2.5	77.6
PVTv2-B1 [97]	13.1	2.1	78.7
Lv-ViT-T [89]	8.5	–	79.1
RegionViT-T [100]	13.8	2.4	80.4
ResNet50 [67]⋆	25.6	4.1	76.1
ResNeXt50-32x4d [244]⋆	25.0	4.3	77.6
RegNetY-4G [86]⋆	21.0	4.0	80.0
EfficientNet-B4 [87]⋆	19.0	4.2	82.9
DeiT-S [12]	22.1	4.6	79.9
PVTv1-S [93]	24.5	3.8	79.8
LocalViT-S [107]	22.4	4.6	80.8
CrossViT-S [104]	26.7	5.6	81.0
TNT-S [88]	23.8	5.2	81.3
Swin-T [36]	29.0	4.5	81.3

方法	参数量（百万）	GFLOPs	Top-1 准确率（%）
NesT-T [111]	17.0	5.8	81.5
T2T-ViTt-14 [35]	21.5	5.2	81.5
CvT-13 [96]	20.0	4.5	81.6
ResT-B [110]	30.3	4.3	81.6
Twins-SVT-S [37]	24.0	2.8	81.7
PVTv2-B2-Li [97]	22.6	3.9	82.1
RegionViT-S [100]	30.6	5.6	82.5
Lv-ViT-S [89]	26.0	6.6	83.3
ResNet101 [67] ⋆	44.7	7.9	77.4
ResNeXt101-32x4d [244]⋆	44.2	8.0	78.8
RegNetY-8G [86]⋆	39.0	8.0	81.7
EfficientNet-B5 [87] ⋆	30.0	9.9	83.6
CvT-21 [96]	32.0	7.1	82.5
CaiT-S-24 [243]	32.2	9.4	82.7
T2T-ViTt-19 [35]	39.0	9.8	81.4
PVTv1-M [93]	44.2	6.7	81.2
PVTv2-B3 [97]	45.2	6.9	83.2
NesT-S [111]	38.0	10.4	83.3
ResNet152 [67] ⋆	60.2	11.6	78.3
CaiT-S-36 [243]	48.0	13.9	83.3
T2T-ViTt-24 [35]	64.0	15.0	82.2
PVTv1-L [93]	61.4	9.8	81.7
TNT-B [88]	66.0	14.1	82.8
Swin-S [36]	50.0	8.7	83.0
Twins-SVT-B [37]	56.0	8.3	83.2
RegionViT-B [100]	72.7	13.0	83.3
PVTv2-B4 [97]	62.6	10.1	83.6
ResNeXt101-64x4d [244] ⋆	83.5	15.6	79.6
RegNetY-16G [86] ⋆	84.0	16.0	82.9
EfficientNet-B6 [87] ⋆	43.0	19.0	84.0
NesT-B [111]	68.0	17.9	83.8
ViT-B/16 [11]	86.6	17.6	79.8
DeiT-B/16 [12]	86.6	17.6	81.8
Swin-B [36]	88.0	15.4	83.3

方法	参数量（百万）	GFLOPs	Top-1 准确率（%）
Twins-SVT-L [37]	99.2	14.8	83.7
PVTv2-B5 [97]	82.0	11.8	83.8
Lv-ViT-M [89]	56.0	16.0	84.1

GFLOPs 按原表作为浮点运算量；⋆ 为纯 CNN 方法。Lv-ViT-T 的 GFLOPs 在原表用破折号表示。

表 3 英文原表 · p.23

Method	#Param (M)	GFLOPs	Top-1 Acc (%)
ResNet18 [67]⋆	11.7	1.8	69.8
EfficientNet-B3 [87]⋆	12.0	1.8	81.6
DeiT-T [12]	5.7	1.3	72.2
T2T-ViT$_t$-7 [35]	5.0	1.3	71.7
LocalViT-T [107]	5.9	1.3	74.8
CrossViT-T [104]	6.9	1.6	73.4
PVTv1-T [93]	13.2	1.9	75.1
ResT-Lite [110]	10.5	1.4	77.2
CaiT-XXX-24 [243]	12.0	2.5	77.6
PVTv2-B1 [97]	13.1	2.1	78.7
Lv-ViT-T [89]	8.5	–	79.1
RegionViT-T [100]	13.8	2.4	80.4
ResNet50 [67]⋆	25.6	4.1	76.1
ResNeXt50-32x4d [244]⋆	25.0	4.3	77.6
RegNetY-4G [86]⋆	21.0	4.0	80.0
EfficientNet-B4 [87]⋆	19.0	4.2	82.9
DeiT-S [12]	22.1	4.6	79.9
PVTv1-S [93]	24.5	3.8	79.8
LocalViT-S [107]	22.4	4.6	80.8
CrossViT-S [104]	26.7	5.6	81.0
TNT-S [88]	23.8	5.2	81.3
Swin-T [36]	29.0	4.5	81.3
NesT-T [111]	17.0	5.8	81.5
T2T-ViT$_t$-14 [35]	21.5	5.2	81.5
CvT-13 [96]	20.0	4.5	81.6
ResT-B [110]	30.3	4.3	81.6
Twins-SVT-S [37]	24.0	2.8	81.7
PVTv2-B2-Li [97]	22.6	3.9	82.1
RegionViT-S [100]	30.6	5.6	82.5
Lv-ViT-S [89]	26.0	6.6	83.3

Method	#Param (M)	GFLOPs	Top-1 Acc (%)
ResNet101 [67] ⋆	44.7	7.9	77.4
ResNeXt101-32x4d [244]⋆	44.2	8.0	78.8
RegNetY-8G [86]⋆	39.0	8.0	81.7
EfficientNet-B5 [87] ⋆	30.0	9.9	83.6
CvT-21 [96]	32.0	7.1	82.5
CaiT-S-24 [243]	32.2	9.4	82.7
T2T-ViT$_t$-19 [35]	39.0	9.8	81.4
PVTv1-M [93]	44.2	6.7	81.2
PVTv2-B3 [97]	45.2	6.9	83.2
NesT-S [111]	38.0	10.4	83.3
ResNet152 [67] ⋆	60.2	11.6	78.3
CaiT-S-36 [243]	48.0	13.9	83.3
T2T-ViT$_t$-24 [35]	64.0	15.0	82.2
PVTv1-L [93]	61.4	9.8	81.7
TNT-B [88]	66.0	14.1	82.8
Swin-S [36]	50.0	8.7	83.0
Twins-SVT-B [37]	56.0	8.3	83.2
RegionViT-B [100]	72.7	13.0	83.3
PVTv2-B4 [97]	62.6	10.1	83.6
ResNeXt101-64x4d [244] ⋆	83.5	15.6	79.6
RegNetY-16G [86] ⋆	84.0	16.0	82.9
EfficientNet-B6 [87] ⋆	43.0	19.0	84.0
NesT-B [111]	68.0	17.9	83.8
ViT-B/16 [11]	86.6	17.6	79.8
DeiT-B/16 [12]	86.6	17.6	81.8
Swin-B [36]	88.0	15.4	83.3
Twins-SVT-L [37]	99.2	14.8	83.7
PVTv2-B5 [97]	82.0	11.8	83.8
Lv-ViT-M [89]	56.0	16.0	84.1

文献 [238] 对视觉 Transformer 的扩展性进行了实证研究，考察了参数量（500 万至 20 亿，即 five million 至 two billion）、训练数据集规模（3000 万至 30 亿张训练图像，即 30 million 至 three billion）以及计算预算（1-10000 TPU 核心日）。由此可得：(a) 扩大计算资源、模型和训练样本规模能提高性能；(b) 只有大型模型（参数较多）能从更多训练数据中获益，小模型的性能很快达到平台期，无法进一步利用额外数据。这表明，大规模模型有能力进一步提升其表征学习能力。不过，在现有架构下扩展 Transformer 既昂贵，又需要难以承受的算力，因此有必要设计更高效的架构。

近期语言领域的研究着力于降低 Transformer 的高复杂度。这种复杂度基本源自自注意力机制 [1]：更新某一词元的表征时，要考虑上一层的所有词元。例如，文献 [103], [245] 探索在更新下一层的各词元时，选择性地或稀疏地关注上一层词元。Linformer [38] 将标准自注意力的复杂度从 $O(n^2)$ 降到 $O(n)$（时间和内存需求均如此），其核心是表明低秩矩阵已足以建模自注意力机制。Reformer [246] 采用局部敏感哈希（LSH），将自注意力复杂度从 $O(n^2)$ 降到 $O(n \log(n))$。为达到类似目的，近期的 Lambda Networks [247] 提议将局部上下文建模为线性函数，以降低自注意力的复杂度。将这些称为 lambda 的线性函数应用于输入查询，以建模像素间的上下文关系。

Vyas 等人 [248] 开发高效的聚类注意力来处理长输入序列，以近似原始的自注意力。聚类注意力先将查询分组，然后计算聚类中心之间的注意力，而不是在所有查询间计算导致二次复杂度的

注意力。其主要思想是，欧氏空间中相近的查询具有相似的注意力分布。聚类数固定时，这一思路使相对于输入序列长度 n 的二次复杂度降为线性复杂度 O(nc)，其中 c 为聚类数。感兴趣的读者可参阅综述 [34]，了解 NLP 中的高效 Transformer。

与 NLP 类似，计算机视觉所用的 Transformer 也受到高计算成本的困扰。例如，基于序列式 Transformer 的图像生成器（如 iGPT）计算开销较高，限制了它们处理高分辨率输入的能力。Transformer 核心自注意力的时间与内存开销随图像块数呈二次增长，即 n 个图像块的开销为 $O(n^2)$；在低层视觉等一些应用中，$n = H \times W$，H 和 W 分别表示图像的高度和宽度。这是现有 Transformer 的一项主要缺点，使其难以用于多数涉及高分辨率（HR）图像的任务，包括高层视觉中的目标检测与分割，以及低层视觉中的超分辨率、去模糊、去噪等。许多方法引入特殊设计，以更“高效”地计算自注意力：例如在自注意力中进行池化/下采样 [97], [219], [249]，用局部窗口注意力 [36], [250]、轴向注意力 [179], [251]、低秩投影注意力 [38], [252], [253]、可核化注意力 [254], [255]，或基于相似性聚类的方法 [246], [256]。然而，这些方案几乎都要在复杂度与准确度间取舍、要求特定的硬件条件，或仍不适用于尺寸很大的图像。因此，迫切需要开发一种高效自注意力机制，在资源有限的系统上处理高分辨率图像而不牺牲准确度。一个值得探索的问题是如何将现有模型扩展到高维情形，例如用一定程度上建模局部上下文的多尺度 Transformer。引入基于视觉任务知识的归纳偏置（如局部邻域空间关系）可以降低高额的计算成本。同样，借助矩阵低秩分解来建模稀疏注意力图，也有助于降低计算开销 [211]。

4.2 数据需求量大

由于 Transformer 架构没有内置用于处理视觉数据的归纳偏置（先验知识），它们通常需要大量训练来弄清相应模态的内在规律。例如，CNN 通过池化运算或多尺度处理模块，具有内在的平移不变性、权重共享和部分尺度不变性；Transformer 网络却需要自行从训练样本中弄清这些图像特有的概念。类似地，自注意力也需要考察大量视频序列，自动发现视频帧之间的关系。结果是训练时间更长、计算需求显著增加，还要处理大型数据集。例如，ViT [11] 需要数以亿计的图像样本，才能在 ImageNet 基准上达到合理性能。如何以较高数据效率训练 Transformer，仍是一个开放问题；近期研究报告了一些令人鼓舞的进展。比如，DeiT [12] 通过蒸馏提高数据效率；T2T（Tokens-to-Token）ViT [35] 则通过合并空间位置相近的词元来建模局部结构，因此仅用 ImageNet 从头训练、不经预训练，也能获得有竞争力的表现。将类似 CNN 的特征层级融入 ViT，以有效捕捉局部图像线索，使 CCT [106]、NesT [111] 等 ViT 甚至可在 CIFAR-10 等小规模数据集上从头训练。文献 [257] 提出另一种提高 ViT 数据效率的方法。作者发现，用锐度感知最小化器（SAM）[258] 平滑局部损失曲面后，ViT 只用简单的数据增强（随机裁剪和水平翻转）[259]，无须计算开销高的强数据增强策略，就可以训练并超过相应的 ResNet。

4.3 面向视觉的 Transformer 设计

我们注意到，现有面向视觉任务的工作，多倾向于直接将 NLP Transformer 用到计算机视觉问题上，包括图像识别 [11]、视频理解 [17]，尤其是多模态处理 [181]。这些直接应用的初步结果颇为鼓舞人心，促使我们进一步研究自注意力和自监督学习的优势，但当前架构可能仍更适合具有序列结构的语言问题；要提高处理视觉输入的效率，还需新的思路。例如，文献 [82] 的向量注意力就是一个很好的尝试：通过学习逐通道注意力，专门使自注意力运算适应视觉输入。类似地，文献 [260] 在 Transformer 的并行分支中使用基于拼图的自监督损失，以改进行人重识别。近期文献 [35] 重新排列空间位置相近的词元，以更好地对邻近位置的关系建模。其他方法还从预训练 CNN 蒸馏词元 [12]，向表征注入领域偏置。有人可能主张，Transformer 等架构应保留通用性，以便直接跨领域使用；但我们注意到，在视觉问题上预训练此类模型需要大量计算和时间，因此需要新的设计策略，使训练更可负担。

4.4 面向 ViT 的神经架构搜索

神经架构搜索（NAS）已被广泛用于为 CNN 寻找优化架构；相对而言，在 Transformer 中的探索还较少，甚至在语言 Transformer 中亦然 [261], [262]。Chen 等人 [263] 提出用于视觉 Transformer 的一次式 NAS，名为 AutoFormer。BossNAS [264] 搜索 CNN 与 Transformer 的混合架构。另一项近期工作 [265] 在视觉应用背景下研究 Transformer 的全局信息与局部信息之间的取舍。进一步用 NAS 探索各领域特有的设计选择，例如语言与视觉领域的不同需求，将有助于像设计 CNN [87] 那样，设计出更高效、更轻量的模型。

4.5 Transformer 的可解释性

Naseer 等人 [266] 通过一系列精心设计的实验，研究了 ViT 在泛化性和鲁棒性方面的一些有趣特性。结果表明，与 CNN 相比，ViT 对纹理变化和严重遮挡具有较强鲁棒性。例如，随机遮挡 80% 的图像内容后，ViT 在 ImageNet 上仍可保持最高达 60% 的 Top-1 准确率。考虑到 Transformer 架构的强劲表现，解释其决策既有意义也很重要，例如直观展示图像中哪些区域与特定分类决策有关。主要难点在于，各层产生的注意力会以复杂方式在后续层中相互混合，导致难以直观展示输入词元对最终预测的相对贡献。这仍是开放问题，不过近期一些工作 [267]–[269] 致力于增强 Transformer 的可解释性，取得了令人鼓舞的结果。文献 [268] 提出注意力展开和注意力流方法，用来更准确地估计注意力。但该方法是经验性的，并作出了简化假设，例如输入词元按跨层的注意力权重作线性组合。Chefer 等人 [269] 指出，无论直接从自注意力（用于编码词元间关系）获取分数，还是如文献 [268] 那样对其重新分配，都不是最佳解法。他们改为在 Transformer 中分配和传播相关性分数，使相关性总量在整个网络中保持不变。该设计能同时处理自注意力层中的正、负归因，还能给出类别特定的可视化。尽管已有这些开创性工作，Transformer 的可视化和解释仍未解决，仍需开发能对特定激活进行空间上精确的可视化的方法。该方向若取得进一步进展，将有助于更好理解 Transformer、诊断其错误行为及决策偏差，也有助于设计规避此类偏差的新架构。

4.6 硬件高效设计

大规模 Transformer 网络可能耗能大、计算需求高，使其难以部署在边缘设备和物联网（IoT）平台等资源受限环境。近期已有工作尝试在 FPGA 等嵌入式系统上压缩和加速 NLP 模型 [270]。Li 等人 [270] 用改进的块循环矩阵表征压缩 NLP 模型，并提出一种新的现场可编程门阵列（FPGA）架构，有效管理资源，以实现高吞吐量和低延迟。以 RoBERTa [7] 为例，与 CPU 相比，其性能（吞吐量，以 FPS 衡量）和能效分别达到 27 倍和 81 倍，功耗降至约三分之一。围绕这一目标，文献 [262] 提议用神经架构搜索策略 [271]–[273] 设计硬件感知 Transformer（HAT）。具体来说，先训练一个 SuperTransformer 来近似估计性能，从而无须完整训练某个模型，便能估计其表现。它包含搜索空间中允许的最大模型，并在公共部分共享权重。最后，考虑硬件延迟约束开展进化搜索，寻找适合目标硬件平台（如物联网设备、GPU 或 CPU）的子模型 SubTransformer。不过，视觉 Transformer 目前缺少这种方便部署到资源受限设备的硬件高效设计。此外，进化算法的搜索成本依然很高，相应的 CO_2 排放会对环境产生影响。

4.7 迈向整合所有模态

由于 Transformer 提供了处理不同模态的统一设计，近期研究也关注用 Transformer 构建适用范围更广的通用推理系统。受能处理多种模态信息的生物系统启发，Perceiver [274] 试图学习一个能处理任意给定输入模态的统一模型，不作特定领域的架构假设。为扩展到高维输入，Perceiver 采用非对称交叉注意力，将输入信息提炼为低维潜在瓶颈特征。将特征提炼成紧凑且维度固定的形式之后，在潜在空间中施加常规 Transformer 模块。原始 Perceiver 的图像分类性能可与 ResNet 和 ViT 竞争，而且能处理三维数据、音频、图像、视频及其组合。但它只能生成固定形式的输出，如类别概率。近期提出的改进 Perceiver IO [275] 旨在学习同时支持灵活输入及任意大小输出的模型，从而能用于自然语言任务、视觉理解等要求结构化输出的问题。这些模型避免了依赖具体模态的架构决策，但学习过程本身仍包含与模态有关的选择，例如特定的数据增强或位置编码。怎样在整个学习流程中实现完全的模态无关性，是未来一个有趣且仍开放的方向。

5 结论

注意力对实现高效、准确的计算机视觉系统起到了关键作用，也为理解深度神经网络的工作方式提供了线索。本文综述了自注意力方法，重点考察建立在自注意力原理上的 Transformer 和双向编码架构。首先讨论自注意力架构的基本概念，然后深入分析多种计算机视觉应用中的代表性竞争方法。具体而言，我们讨论了图像识别、目标检测、语义与实例分割、视频分析与分类、视觉问答、视觉常识推理、图像描述生成、视觉–语言导航、聚类、小样本学习及三维数据分析中的当时最先进的自注意力模型。我们系统指出了现有方法的主要优势和局限，着重阐述了未来的重要研究方向。聚焦计算机视觉任务，使本文能从专门的视角呈现自注意力和 Transformer 方法的近期进展。我们希望这项工作能引起视觉研究界进一步关注，充分发挥 Transformer 模型的潜力，并改善当前的不足，例如减少其碳足迹。

致谢

作者感谢 Tim Prangemeier（达姆施塔特工业大学）、Luowei Zhou（微软研究院）、Jason Corso（密歇根大学）、Pichao Wang（阿里巴巴集团）、Yuqing Wang（美团）、Alex Meinke（图宾根大学）、Irwan Bello（Google Brain）和 Manoj Kumar（Google Brain）对本综述提供的有益反馈。我们也感谢 Mohamed Afham 协助制作一幅图。

参考文献

保留英文原文的书目与编号。

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in NeurIPS, 2017.

[2] M. Ott, S. Edunov, D. Grangier, and M. Auli, “Scaling neural machine translation,” in WMT, 2018.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pretraining of deep bidirectional transformers for language understanding,” arXiv preprint arXiv:1810.04805, 2018.

[4] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” tech. rep., OpenAI, 2018.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” tech. rep., OpenAI, 2019.

[6] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., “Language models are few-shot learners,” arXiv preprint arXiv:2005.14165, 2020.

[7] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “RoBERTa: A robustly optimized bert pretraining approach,” arXiv preprint arXiv:1907.11692, 2019.

[8] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” arXiv preprint arXiv:1910.10683, 2019.

[9] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, and Z. Chen, “Gshard: Scaling giant models with conditional computation and automatic sharding,” arXiv preprint arXiv:2006.16668, 2020.

[10] W. Fedus, B. Zoph, and N. Shazeer, “Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity,” arXiv preprint arXiv:2101.03961.

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” arXiv preprint arXiv:2010.11929, 2020.

[12] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, “Training data-efficient image transformers & distillation through attention,” arXiv preprint arXiv:2012.12877, 2020.

[13] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” arXiv preprint arXiv:2005.12872, 2020.

[14] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, “Deformable DETR: Deformable transformers for end-to-end object detection,” arXiv preprint arXiv:2010.04159, 2020.

[15] L. Ye, M. Rochan, Z. Liu, and Y. Wang, “Cross-modal self-attention network for referring image segmentation,” in CVPR, 2019.

[16] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, “Learning texture transformer network for image super-resolution,” in CVPR, 2020.

[17] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, “VideoBERT: A joint model for video and language representation learning,” in ICCV, 2019.

[18] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, “Video action transformer network,” in CVPR, 2019.

[19] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, and W. Gao, “Pre-trained image processing transformer,” arXiv preprint arXiv:2012.00364, 2020.

[20] A. Ramesh, M. Pavlov, G. Goh, and S. Gray, “DALL·E: Creating images from text,” tech. rep., OpenAI, 2021.

[21] H. Tan and M. Bansal, “LXMERT: Learning cross-modality encoder representations from transformers,” in EMNLP-IJCNLP, 2019.

[22] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, “VL-BERT: Pre-training of generic visual-linguistic representations,” arXiv preprint arXiv:1908.08530, 2019.

[23] X. Wang, C. Yeshwanth, and M. Nießner, “SceneFormer: Indoor scene generation with transformers,” arXiv preprint arXiv:2012.09793, 2020.

[24] M. Kumar, D. Weissenborn, and N. Kalchbrenner, “Colorization transformer,” in ICLR, 2021.

[25] C. Doersch, A. Gupta, and A. Zisserman, “CrossTransformers: spatially-aware few-shot transfer,” NeurIPS, 2020.

[26] H.-J. Ye, H. Hu, D.-C. Zhan, and F. Sha, “Few-shot learning via embedding adaptation with set-to-set functions,” in CVPR, 2020.

[27] S. Chaudhari, G. Polatkan, R. Ramanath, and V. Mithal, “An attentive survey of attention models,” arXiv preprint arXiv:1904.02874, 2019.

[28] A. de Santana Correia and E. L. Colombini, “Attention, please! asurvey of neural attention models in deep learning,” arXiv preprint arXiv:2103.16775, 2021.

[29] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and tell: A neural image caption generator,” in CVPR, 2015.

[30] Y. Bengio, I. Goodfellow, and A. Courville, Deep learning. MIT press, 2017.

[31] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, 2015.

[32] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, 1997.

[33] D. Hu, “An introductory survey on attention mechanisms in nlp problems,” in IntelliSys, 2019.

[34] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, “Efficient transformers: A survey,” arXiv preprint arXiv:2009.06732, 2020.

[35] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, F. E. Tay, J. Feng, and S. Yan, “Tokens-to-token vit: Training vision transformers from scratch on imagenet,” arXiv preprint arXiv:2101.11986, 2021.

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

[37] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, and C. Shen, “Twins: Revisiting the design of spatial attention in vision transformers,” arXiv preprint arXiv:2104.13840, 2021.

[38] S. Wang, B. Li, M. Khabsa, H. Fang, and H. Ma, “Linformer: Self-attention with linear complexity,” arXiv preprint arXiv:2006.04768, 2020.

[39] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-attention generative adversarial networks,” in International conference on machine learning, pp. 7354–7363, PMLR, 2019.

[40] J. Pérez, J. Marinković, and P. Barceló, “On the turing completeness of modern neural network architectures,” in International Conference on Learning Representations, 2018.

[41] J.-B. Cordonnier, A. Loukas, and M. Jaggi, “On the relationship between self-attention and convolutional layers,” in International Conference on Learning Representations, 2019.

[42] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, “Deformable convolutional networks,” in Proceedings of the IEEE international conference on computer vision, pp. 764–773, 2017.

[43] Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu, “UNITER: Universal image-text representation learning,” in ECCV, 2020.

[44] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, et al., “Oscar: Object-semantics aligned pretraining for vision-language tasks,” in ECCV, 2020.

[45] K. Lin, L. Wang, and Z. Liu, “End-to-end human pose and mesh reconstruction with transformers,” arXiv preprint arXiv:2012.09760, 2020.

[46] S. Gidaris, P. Singh, and N. Komodakis, “Unsupervised representation learning by predicting image rotations,” in ICLR, 2018.

[47] “Revisiting the unreasonable effectiveness of data.” https://ai.googleblog.com/2017/07/revisiting-unreasonable-effectiveness.html. Accessed: 2020-12-31.

[48] L. Jing and Y. Tian, “Self-supervised visual feature learning with deep neural networks: A survey,” TPAMI, 2020.

[49] X. Liu, F. Zhang, Z. Hou, Z. Wang, L. Mian, J. Zhang, and J. Tang, “Self-supervised learning: Generative or contrastive,” arXiv preprint arXiv:2006.08218, 2020.

[50] “Aaai 2020 keynotes turing award winners event.”

https://www.youtube.com/watch?v=UX8OubxsY8w. Accessed: 2020-12-31.

[51] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” in ECCV, 2016.

[52] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., “Photo-realistic single image super-resolution using a generative adversarial network,” in CVPR, 2017.

[53] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. Efros, “Context encoders: Feature learning by inpainting,” in CVPR, 2016.

[54] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in NeurIPS, 2014.

[55] D. Lin, K. Fu, Y. Wang, G. Xu, and X. Sun, “MARTA GANs: Unsupervised representation learning for remote sensing image classification,” GRSL, 2017.

[56] U. Ahsan, R. Madhok, and I. Essa, “Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition,” in WACV, 2019.

[57] M. Noroozi and P. Favaro, “Unsupervised learning of visual representations by solving jigsaw puzzles,” in ECCV, 2016.

[58] D. Kim, D. Cho, D. Yoo, and I. S. Kweon, “Learning image representations by completing damaged jigsaw puzzles,” WACV, 2018.

[59] L. Jing, X. Yang, J. Liu, and Y. Tian, “Self-supervised spatiotemporal feature learning via video rotation prediction,” arXiv preprint arXiv:1811.11387, 2018.

[60] H.-Y. Lee, J.-B. Huang, M. Singh, and M.-H. Yang, “Unsupervised representation learning by sorting sequences,” in ICCV, 2017.

[61] I. Misra, C. L. Zitnick, and M. Hebert, “Shuffle and learn: unsupervised learning using temporal order verification,” in ECCV, 2016.

[62] D. Wei, J. J. Lim, A. Zisserman, and W. T. Freeman, “Learning and using the arrow of time,” in CVPR, 2018.

[63] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, “VisualBERT: A simple and performant baseline for vision and language,” in Arxiv preprint arXiv:1908.03557, 2019.

[64] B. Korbar, D. Tran, and L. T., “Cooperative learning of audio and video models from self-supervised synchronization,” in NeurIPS, 2018.

[65] R. Arandjelovic and A. Zisserman, “Look, listen and learn,” in ICCV, 2017.

[66] N. Sayed, B. Brattoli, and B. Ommer, “Cross and learn: Cross-modal self-supervision,” in GCPR, 2018.

[67] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in CVPR, 2016.

[68] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” arXiv preprint arXiv:1607.06450, 2016.

[69] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in CVPR, 2005.

[70] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in CVPR, 2018.

[71] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al., “The kinetics human action video dataset,” arXiv preprint arXiv:1705.06950, 2017.

[72] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, “CCNet: Criss-cross attention for semantic segmentation,” in ICCV, 2019.

[73] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The cityscapes dataset for semantic urban scene understanding,” in CVPR, 2016.

[74] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, “Scene parsing through ade20k dataset,” in CVPR, 2017.

[75] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in ECCV, 2014.

[76] X. Liang, K. Gong, X. Shen, and L. Lin, “Look into person: Joint body parsing & pose estimation network and a new benchmark,” TPAMI, 2018.

[77] G. J. Brostow, J. Fauqueur, and R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” Pattern Recognition Letters, 2009.

[78] H. Hu, Z. Zhang, Z. Xie, and S. Lin, “Local relation networks for image recognition,” in ICCV, 2019.

[79] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le, “Attention augmented convolutional networks,” in ICCV, 2019.

[80] P. Shaw, J. Uszkoreit, and A. Vaswani, “Self-attention with relative position representations,” in NAACL, 2018.

[81] N. Parmar, P. Ramachandran, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, “Stand-alone self-attention in vision models,” in NeurIPS, 2019.

[82] H. Zhao, J. Jia, and V. Koltun, “Exploring self-attention for image recognition,” in CVPR, 2020.

[83] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” arXiv preprint arXiv:1312.6199, 2013.

[84] M. M. Naseer, S. H. Khan, M. H. Khan, F. S. Khan, and F. Porikli, “Cross-domain transferability of adversarial perturbations,” in NeurIPS, 2019.

[85] M. Naseer, K. Ranasinghe, S. Khan, F. S. Khan, and F. Porikli, “On improving adversarial transferability of vision transformers,” arXiv preprint arXiv:2106.04169, 2021.

[86] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, “Designing network design spaces,” in CVPR, 2020.

[87] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in ICML, 2019.

[88] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, “Transformer in transformer,” arXiv preprint arXiv:2103.00112, 2021.

[89] Z. Jiang, Q. Hou, L. Yuan, D. Zhou, Y. Shi, X. Jin, A. Wang, and J. Feng, “All tokens matter: Token labeling for training better vision transformers,” arXiv preprint arXiv:2104.10858, 2021.

[90] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032, 2019.

[91] A. El-Nouby, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, and H. Jegou, “Xcit: Cross-covariance image transformers,” 2021.

[92] D. Zhou, B. Kang, X. Jin, L. Yang, X. Lian, Z. Jiang, Q. Hou, and J. Feng, “Deepvit: Towards deeper vision transformer,” 2021.

[93] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, “Pyramid vision transformer: A versatile backbone for dense prediction without convolutions,” arXiv preprint arXiv:2102.12122, 2021.

[94] J. Yang, C. Li, P. Zhang, X. Dai, B. Xiao, L. Yuan, and J. Gao, “Focal self-attention for local-global interactions in vision transformers,” 2021.

[95] Z. Huang, Y. Ben, G. Luo, P. Cheng, G. Yu, and B. Fu, “Shuffle transformer: Rethinking spatial shuffle for vision transformer,” 2021.

[96] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, “Cvt: Introducing convolutions to vision transformers,” arXiv preprint arXiv:2103.15808, 2021.

[97] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, “Pvtv2: Improved baselines with pyramid vision transformer,” 2021.

[98] W. Xu, Y. Xu, T. Chang, and Z. Tu, “Co-scale conv-attentional image transformers,” 2021.

[99] W. Wang, L. Yao, L. Chen, D. Cai, X. He, and W. Liu, “Crossformer: A versatile vision transformer based on cross-scale attention,” arXiv preprint arXiv:2108.00154, 2021.

[100] C.-F. Chen, R. Panda, and Q. Fan, “Regionvit: Regional-to-local attention for vision transformers,” 2021.

[101] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, “Segformer: Simple and efficient design for semantic segmentation with transformers,” 2021.

[102] P. Zhang, X. Dai, J. Yang, B. Xiao, L. Yuan, L. Zhang, and J. Gao, “Multi-scale vision longformer: A new vision transformer for high-resolution image encoding,” ICCV 2021, 2021.

[103] I. Beltagy, M. E. Peters, and A. Cohan, “Longformer: The long-document transformer,” arXiv preprint arXiv:2004.05150, 2020.

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

[105] K. Yuan, S. Guo, Z. Liu, A. Zhou, F. Yu, and W. Wu, “Incorporating convolution designs into visual transformers,” arXiv preprint arXiv:2103.11816, 2021.

[106] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, and H. Shi, “Escaping the big data paradigm with compact transformers,” 2021.

[107] Y. Li, K. Zhang, J. Cao, R. Timofte, and L. V. Gool, “Localvit: Bringing locality to vision transformers,” 2021.

[108] B. Graham, A. El-Nouby, H. Touvron, P. Stock, A. Joulin, H. Jégou, and M. Douze, “Levit: a vision transformer in convnet’s clothing for faster inference,” 2021.

[109] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Backpropagation applied to handwritten zip code recognition,” Neural computation, vol. 1, no. 4, pp. 541–551, 1989.

[110] Q. Zhang and Y. Yang, “Rest: An efficient transformer for visual recognition,” arXiv preprint arXiv:2105.13677, 2021.

[111] Z. Zhang, H. Zhang, L. Zhao, T. Chen, and T. Pfister, “Aggregating nested transformers,” in arXiv preprint arXiv:2105.12723, 2021.

[112] Z. Dai, H. Liu, Q. V. Le, and M. Tan, “Coatnet: Marrying convolution and attention for all data sizes,” 2021.

[113] X. Chu, Z. Tian, B. Zhang, X. Wang, X. Wei, H. Xia, and C. Shen, “Conditional positional encodings for vision transformers,” 2021.

[114] Y. Liu, G. Sun, Y. Qiu, L. Zhang, A. Chhatkuli, and L. Van Gool, “Transformer in convolutional neural networks,” arXiv preprint arXiv:2106.03180, 2021.

[115] X. Chen, S. Xie, and K. He, “An empirical study of training self-supervised visual transformers,” arXiv e-prints, pp. arXiv–2104, 2021.

[116] X. Chen, H. Fan, R. Girshick, and K. He, “Improved baselines with momentum contrastive learning,” arXiv preprint arXiv:2003.04297, 2020.

[117] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9729–9738, 2020.

[118] Z. Xie, Y. Lin, Z. Yao, Z. Zhang, Q. Dai, Y. Cao, and H. Hu, “Self-supervised learning with swin transformers,” arXiv preprint arXiv:2105.04553, 2021.

[119] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, et al., “Bootstrap your own latent: A new approach to self-supervised learning,” arXiv preprint arXiv:2006.07733, 2020.

[120] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, “Emerging properties in self-supervised vision transformers,” arXiv preprint arXiv:2104.14294, 2021.

[121] C. Li, J. Yang, P. Zhang, M. Gao, B. Xiao, X. Dai, L. Yuan, and J. Gao, “Efficient self-supervised vision transformers for representation learning,” arXiv preprint arXiv:2106.09785, 2021.

[122] Y. Wang, X. Zhang, T. Yang, and J. Sun, “Anchor detr: Query design for transformer-based detector,” 2021.

[123] T. Chen, S. Saxena, L. Li, D. J. Fleet, and G. Hinton, “Pix2seq: A language modeling framework for object detection,” 2021.

[124] Y. Fang, B. Liao, X. Wang, J. Fang, J. Qi, R. Wu, J. Niu, and W. Liu, “You only look at one sequence: Rethinking transformer in vision through object detection,” 2021.

[125] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” TPAMI, 2016.

[126] R. Girshick, “Fast R-CNN,” in ICCV, 2015.

[127] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in ICCV, 2017.

[128] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in CVPR, 2016.

[129] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in ECCV, 2016.

[130] T. Prangemeier, C. Reich, and H. Koeppl, “Attention-based transformers for instance segmentation of cells in microstructures,” in 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 700–707, IEEE, 2020.

[131] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in CVPR, 2017.

[132] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” 2020.

[133] H. Wang, Y. Zhu, B. Green, H. Adam, A. Yuille, and L.-C. Chen, “Axial-DeepLab: Stand-alone axial-attention for panoptic segmentation,” arXiv preprint arXiv:2003.07853, 2020.

[134] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. S. Torr, and L. Zhang, “Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers,” 2021.

[135] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, “Segmenter: Transformer for semantic segmentation,” 2021.

[136] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, “Panoptic segmentation,” in CVPR, 2019.

[137] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, “The mapillary vistas dataset for semantic understanding of street scenes,” in ICCV, 2017.

[138] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in CVPR, 2009.

[139] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, “Modeling context in referring expressions,” in ECCV, 2016.

[140] J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy, “Generation and comprehension of unambiguous object descriptions,” in CVPR, 2016.

[141] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, “Referitgame: Referring to objects in photographs of natural scenes,” in EMNLP, 2014.

[142] N. Parmar, A. Vaswani, J. Uszkoreit, Ł. Kaiser, N. Shazeer, A. Ku, and D. Tran, “Image transformer,” in ICML, 2018.

[143] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever, “Generative pretraining from pixels,” in ICML, 2020.

[144] P. Esser, R. Rombach, and B. Ommer, “Taming transformers for high-resolution image synthesis,” arXiv:2012.09841, 2020.

[145] Y. Jiang, S. Chang, and Z. Wang, “Transgan: Two transformers can make one strong gan,” 2021.

[146] A. K. Bhunia, S. Khan, H. Cholakkal, R. M. Anwer, F. S. Khan, and M. Shah, “Handwriting transformers,” arXiv preprint arXiv:2104.03964, 2021.

[147] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al., “Conditional image generation with pixelcnn decoders,” in NeurIPS, 2016.

[148] A. Krizhevsky, “Learning multiple layers of features from tiny images,” tech. rep., 2009.

[149] A. Coates, A. Ng, and H. Lee, “An analysis of single-layer networks in unsupervised feature learning,” in AISTATS, 2011.

[150] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” arXiv preprint arXiv:2002.05709, 2020.

[151] P. Bachman, R. Hjelm, and W. Buchwalter, “Learning representations by maximizing mutual information across views,” in NeurIPS, 2019.

[152] O. J. Hénaff, A. Srinivas, J. De Fauw, A. Razavi, C. Doersch, S. Eslami, and A. v. d. Oord, “Data-efficient image recognition with contrastive predictive coding,” arXiv preprint arXiv:1905.09272, 2019.

[153] Y. Tian, D. Krishnan, and P. Isola, “Contrastive multiview coding,” arXiv preprint arXiv:1906.05849, 2019.

[154] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, “A guide to convolutional neural networks for computer vision,” Synthesis Lectures on Computer Vision, 2018.

[155] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” arXiv preprint arXiv:1511.06434, 2015.

[156] C. Gao, Y. Chen, S. Liu, Z. Tan, and S. Yan, “Adversarialnas: Adversarial neural architecture search for gans,” in CVPR, pp. 5680– 5689, 2020.

[157] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” in CVPR, pp. 8110–8119, 2020.

[158] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative adversarial text to image synthesis,” in ICML, 2016.

[159] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, “StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks,” in ICCV, 2017.

[160] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, “StackGAN++: Realistic image synthesis with stacked generative adversarial networks,” TPAMI, 2018.

[161] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, “AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks,” in CVPR, 2018.

[162] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” arXiv preprint arXiv:1312.6114, 2013.

[163] A. Razavi, A. van den Oord, and O. Vinyals, “Generating diverse high-fidelity images with vq-vae-2,” in NeurISP, 2019.

[164] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, “Swinir: Image restoration using swin transformer,” in ICCVW, 2021.

[165] Z. Wang, X. Cun, J. Bao, and J. Liu, “Uformer: A general u-shaped transformer for image restoration,” arXiv preprint arXiv:2106.03106, 2021.

[166] Z. Lu, H. Liu, J. Li, and L. Zhang, “Efficient transformer for single image super-resolution,” arXiv preprint arXiv:2108.11084, 2021.

[167] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, “Image super-resolution using very deep residual channel attention networks,” in ECCV, 2018.

[168] T. Dai, J. Cai, Y. Zhang, S. Xia, and L. Zhang, “Second-order attention network for single image super-resolution,” in CVPR, 2019.

[169] B. Niu, W. Wen, W. Ren, X. Zhang, L. Yang, S. Wang, K. Zhang, X. Cao, and H. Shen, “Single image super-resolution via a holistic attention network,” in ECCV, 2020.

[170] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, “Enhanced deep residual networks for single image super-resolution,” in CVPRW, 2017.

[171] Y. Tai, J. Yang, and X. Liu, “Image super-resolution via deep recursive residual network,” in CVPR, 2017.

[172] W. Han, S. Chang, D. Liu, M. Yu, M. Witbrock, and T. Huang, “Image super-resolution via dual-state recurrent networks,” in CVPR, 2018.

[173] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, “Residual dense network for image restoration,” TPAMI, 2020.

[174] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, “ESRGAN: enhanced super-resolution generative adversarial networks,” in ECCVW, 2018.

[175] S.-J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee, “SRFEAT: Single image super-resolution with feature discrimination,” in ECCV, 2018.

[176] M. S. Sajjadi, B. Scholkopf, and M. Hirsch, “EnhanceNet: Single image super-resolution through automated texture synthesis,” in ICCV, 2017.

[177] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., “Photo-realistic single image super-resolution using a generative adversarial network,” in CVPR, 2017.

[178] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in ECCV, 2016.

[179] J. Ho, N. Kalchbrenner, D. Weissenborn, and T. Salimans, “Axial attention in multidimensional transformers,” arXiv preprint arXiv:1912.12180, 2019.

[180] G. Li, N. Duan, Y. Fang, M. Gong, D. Jiang, and M. Zhou, “Unicoder-VL: A universal encoder for vision and language by cross-modal pre-training.,” in AAAI, 2020.

[181] J. Lu, D. Batra, D. Parikh, and S. Lee, “Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks,” in NeurIPS, 2019.

[182] S. Lee, Y. Yu, G. Kim, T. Breuel, J. Kautz, and Y. Song, “Parameter efficient multimodal transformers for video representation learning,” arXiv preprint arXiv:2012.04124, 2020.

[183] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, “VQA: Visual question answering,” in ICCV, 2015.

[184] R. Zellers, Y. Bisk, A. Farhadi, and Y. Choi, “From recognition to cognition: Visual commonsense reasoning,” in CVPR, 2019.

[185] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He, “Stacked cross attention for image-text matching,” in ECCV, 2018.

[186] A. Suhr, S. Zhou, A. Zhang, I. Zhang, H. Bai, and Y. Artzi, “A corpus for reasoning about natural language grounded in photographs,” arXiv preprint arXiv:1811.00491, 2018.

[187] J. Carreira, E. Noland, C. Hillier, and A. Zisserman, “A short note on the kinetics-700 human action dataset,” arXiv:1907.06987, 2019.

[188] K. Soomro, A. R. Zamir, and M. Shah, “UCF101: A dataset of 101 human actions classes from videos in the wild,” arXiv preprint arXiv:1212.0402, 2012.

[189] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in ICASSP, 2017.

[190] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, “Hollywood in homes: Crowdsourcing data collection for activity understanding,” in ECCV, 2016.

[191] H. Tan and M. Bansal, “Vokenization: Improving language understanding with contextualized, visual-grounded supervision,” in EMNLP, 2020.

[192] W. Hao, C. Li, X. Li, L. Carin, and J. Gao, “Towards learning a generic agent for vision-and-language navigation via pretraining,” in CVPR, 2020.

[193] A. Majumdar, A. Shrivastava, S. Lee, P. Anderson, D. Parikh, and D. Batra, “Improving vision-and-language navigation with image-text pairs from the web,” arXiv preprint arXiv:2004.14973, 2020.

[194] K. Chen, J. K. Chen, J. Chuang, M. Vázquez, and S. Savarese, “Topological planning with transformers for vision-and-language navigation,” arXiv preprint arXiv:2012.05292, 2020.

[195] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., “Learning transferable visual models

from natural language supervision," Image, vol. 2, p. T2, 2021.

[196] P. Sharma, N. Ding, S. Goodman, and R. Soricut, "Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning," in ACL, 2018.

[197] L. Zhou, H. Palangi, L. Zhang, H. Hu, J. Corso, and J. Gao, "Unified vision-language pre-training for image captioning and vqa," in AAAI, vol. 34, pp. 13041–13049, 2020.

[198] C. Sun, F. Baradel, K. Murphy, and C. Schmid, "Learning video representations using contrastive bidirectional transformer," arXiv preprint arXiv:1906.05743, 2019.

[199] C. Alberti, J. Ling, M. Collins, and D. Reitter, "Fusion of detected objects in text for visual question answering," in EMNLP, 2019.

[200] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al., "Visual genome: Connecting language and vision using crowdsourced dense image annotations," IJCV, 2017.

[201] V. Ordonez, G. Kulkarni, and T. L. Berg, "Im2text: Describing images using 1 million captioned photographs," in NeurIPS, 2011.

[202] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, "Vinvl: Revisiting visual representations in vision-language models," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5579–5588, 2021.

[203] A. Kamath, M. Singh, Y. LeCun, I. Misra, G. Synnaeve, and N. Carion, "Mdetr–modulated detection for end-to-end multimodal understanding," arXiv preprint arXiv:2104.12763, 2021.

[204] J. Deng, Z. Yang, T. Chen, W. Zhou, and H. Li, "Transvg: End-to-end visual grounding with transformers," 2021.

[205] M. Li and L. Sigal, "Referring transformer: A one-step approach to multi-task visual grounding," arXiv preprint arXiv:2106.03089, 2021.

[206] Y. Du, Z. Fu, Q. Liu, and Y. Wang, "Visual grounding with transformers," arXiv preprint arXiv:2105.04281, 2021.

[207] S. Ging, M. Zolfaghari, H. Pirsiavash, and T. Brox, "COOT: Cooperative hierarchical transformer for video-text representation learning," arXiv preprint arXiv:2011.00597, 2020.

[208] H. Seong, J. Hyun, and E. Kim, "Video multitask transformer network," in ICCV Workshops, pp. 0–0, 2019.

[209] Y. Wang, Z. Xu, X. Wang, C. Shen, B. Cheng, H. Shen, and H. Xia, "End-to-end video instance segmentation with transformers," arXiv preprint arXiv:2011.14503, 2020.

[210] L. Zhou, Y. Zhou, J. Corso, R. Socher, and C. Xiong, "End-to-end dense video captioning with masked transformer," in CVPR, 2018.

[211] D. Neimark, O. Bar, M. Zohar, and D. Asselmann, "Video transformer network," arXiv preprint arXiv:2102.00719, 2021.

[212] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," arXiv preprint arXiv:2103.15691, 2021.

[213] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?," in Proceedings of the International Conference on Machine Learning (ICML), July 2021.

[214] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, "Dense-captioning events in videos," in ICCV, pp. 706–715, 2017.

[215] L. Zhou, C. Xu, and J. Corso, "Towards automatic learning of procedures from web instructional videos," in AAAI, vol. 32, 2018.

[216] C. Plizzari, M. Cannici, and M. Matteucci, "Spatial temporal transformer network for skeleton-based action recognition," arXiv preprint arXiv:2008.07404, 2020.

[217] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3d human activity analysis," in CVPR, 2016.

[218] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "NTU RGB+D 120: A large-scale benchmark for 3d human activity understanding," TPAMI, 2019.

[219] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," 2021.

[220] J. Wang, G. Bertasius, D. Tran, and L. Torresani, "Long-short temporal contrastive learning of video transformers," arXiv preprint arXiv:2106.09212, 2021.

[221] L. Yang, Y. Fan, and N. Xu, "Video instance segmentation," in ICCV, pp. 5188–5197, 2019.

[222] G. Bertasius and L. Torresani, "Classifying, segmenting, and tracking object instances in video with mask propagation," in CVPR, pp. 9739–9748, 2020.

[223] E. Triantafillou, T. Zhu, V. Dumoulin, P. Lamblin, U. Evci, K. Xu, R. Goroshin, C. Gelada, K. Swersky, P.-A. Manzagol, et al., "Meta-dataset: A dataset of datasets for learning to learn from few examples," in ICLR, 2020.

[224] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907, 2016.

[225] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola, "Deep sets," in NeurIPS, 2017.

[226] L. Liu, W. Hamilton, G. Long, J. Jiang, and H. Larochelle, "A universal representation transformer layer for few-shot image classification," 2020.

[227] H. Edwards and A. Storkey, "Towards a neural statistician," arXiv preprint arXiv:1606.02185, 2016.

[228] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh, "Set transformer: A framework for attention-based permutation-invariant neural networks," in ICML, 2019.

[229] J. Lee, Y. Lee, and Y. W. Teh, "Deep amortized clustering," arXiv preprint arXiv:1909.13433, 2019.

[230] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, "Point transformer," arXiv preprint arXiv:2012.09164, 2020.

[231] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "Pct: Point cloud transformer," arXiv preprint arXiv:2012.09688, 2020.

[232] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in CVPR, 2015.

[233] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3d model repository," arXiv preprint arXiv:1512.03012, 2015.

[234] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments," TPAMI, 2013.

[235] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3d human pose in the wild using imus and a moving camera," in ECCV, 2018.

[236] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox, "FreiHAND: A dataset for markerless capture of hand pose and shape from single rgb images," in ICCV, 2019.

[237] "OpenAI's GPT-3 language model: A technical overview." https://lambdalabs.com/blog/demystifying-gpt-3/. Accessed: 2020-12-31.

[238] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," 2021.

[239] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," TACL, 2014.

[240] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the v in vqa matter: Elevating the role of image understanding in visual question answering," in CVPR, 2017.

[241] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in ICCV, 2015.

[242] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," NeurIPS, 2017.

[243] H. Touvron, M. Cord, A. Sablayrolles, G. Synnaeve, and H. Jégou, "Going deeper with image transformers," arXiv preprint arXiv:2103.17239, 2021.

[244] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in CVPR, 2017.

[245] R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," arXiv:1904.10509, 2019.

[246] N. Kitaev, Ł. Kaiser, and A. Levskaya, "Reformer: The efficient transformer," in ICLR, 2020.

[247] I. Bello, "Lambdanetworks: Modeling long-range interactions without attention," in International Conference on Learning Representations, 2021.

[248] A. Vyas, A. Katharopoulos, and F. Fleuret, "Fast transformers with clustered attention," NeurIPS, 2020.

[249] Y.-H. Wu, Y. Liu, X. Zhan, and M.-M. Cheng, "P2t: Pyramid pooling transformer for scene understanding," arXiv preprint arXiv:2106.12011, 2021.

[250] A. Vaswani, P. Ramachandran, A. Srinivas, N. Parmar, B. Hechtman, and J. Shlens, "Scaling local self-attention for parameter efficient visual backbones," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12894–12904, 2021.

[251] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, L. Yuan, D. Chen, and B. Guo, “Cswin transformer: A general vision transformer backbone with cross-shaped windows,” arXiv preprint arXiv:2107.00652, 2021.

[252] Y. Xiong, Z. Zeng, R. Chakraborty, M. Tan, G. Fung, Y. Li, and V. Singh, “Nyströmformer: A nyström-based algorithm for approximating self-attention,” in AAAI, 2021.

[253] Y. Tay, D. Bahri, D. Metzler, D. Juan, Z. Zhao, and C. Zheng, “Synthesizer: Rethinking self-attention in transformer models,” in ICML, 2021.

[254] H. Peng, N. Pappas, D. Yogatama, R. Schwartz, N. A. Smith, and L. Kong, “Random feature attention,” in ICLR, 2021.

[255] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, et al., “Rethinking attention with performers,” in ICLR, 2021.

[256] Y. Tay, D. Bahri, L. Yang, D. Metzler, and D.-C. Juan, “Sparse sinkhorn attention,” in ICML, 2020.

[257] X. Chen, C.-J. Hsieh, and B. Gong, “When vision transformers outperform resnets without pretraining or strong data augmentations,” arXiv preprint arXiv:2106.01548, 2021.

[258] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, “Sharpness-aware minimization for efficiently improving generalization,” arXiv preprint arXiv:2010.01412, 2020.

[259] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818–2826, 2016.

[260] S. He, H. Luo, P. Wang, F. Wang, H. Li, and W. Jiang, “Transreid: Transformer-based object re-identification,” arXiv:2102.04378, 2021.

[261] D. R. So, C. Liang, and Q. V. Le, “The evolved transformer,” 2019.

[262] H. Wang, Z. Wu, Z. Liu, H. Cai, L. Zhu, C. Gan, and S. Han, “Hat: Hardware-aware transformers for efficient natural language processing,” 2020.

[263] M. Chen, H. Peng, J. Fu, and H. Ling, “Autoformer: Searching transformers for visual recognition,” arXiv preprint arXiv:2107.00651, 2021.

[264] C. Li, T. Tang, G. Wang, J. Peng, B. Wang, X. Liang, and X. Chang, “Bossnas: Exploring hybrid cnn-transformers with block-wisely self-supervised neural architecture search,” arXiv preprint arXiv:2103.12424, 2021.

[265] B. Chen, P. Li, C. Li, B. Li, L. Bai, C. Lin, M. Sun, W. Ouyang, et al., “Glit: Neural architecture search for global and local image transformer,” arXiv preprint arXiv:2107.02960, 2021.

[266] M. Naseer, K. Ranasinghe, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, “Intriguing properties of vision transformers,” arXiv preprint arXiv:2105.10497, 2021.

[267] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, “Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned,” arXiv preprint arXiv:1905.09418, 2019.

[268] S. Abnar and W. Zuidema, “Quantifying attention flow in transformers,” arXiv preprint arXiv:2005.00928, 2020.

[269] H. Chefer, S. Gur, and L. Wolf, “Transformer interpretability beyond attention visualization,” arXiv preprint arXiv:2012.09838, 2020.

[270] B. Li, S. Pandey, H. Fang, Y. Lyv, J. Li, J. Chen, M. Xie, L. Wan, H. Liu, and C. Ding, “FTRANS: energy-efficient acceleration of transformers using fpga,” in ISLPED, 2020.

[271] G. Bender, P.-J. Kindermans, B. Zoph, V. Vasudevan, and Q. Le, “Understanding and simplifying one-shot architecture search,” in ICML, 2018.

[272] Z. Guo, X. Zhang, H. Mu, W. Heng, Z. Liu, Y. Wei, and J. Sun, “Single path one-shot neural architecture search with uniform sampling,” arXiv preprint arXiv:1904.00420, 2019.

[273] H. Pham, M. Y. Guan, B. Zoph, Q. V. Le, and J. Dean, “Efficient neural architecture search via parameter sharing,” in ICML, 2018.

[274] A. Jaegle, F. Gimeno, A. Brock, A. Zisserman, O. Vinyals, and J. Carreira, “Perceiver: General perception with iterative attention,” arXiv preprint arXiv:2103.03206, 2021.

[275] A. Jaegle, S. Borgeaud, J.-B. Alayrac, C. Doersch, C. Ionescu, D. Ding, S. Koppula, D. Zoran, A. Brock, E. Shelhamer, et al., “Perceiver io: A general architecture for structured inputs & outputs,” arXiv preprint arXiv:2107.14795, 2021.