中图分类号: 学校代码: 10055
UDC: 密级: 公开

南開大學

硕 士 学 位 论 文

基于卷积注意力机制的视觉识别和语义分割骨干网络设计

Designing Backbone Networks for Visual Recognition and Semantic Segmentation with Convolutional Attention Mechanism

论文作者 陆承泽 指导教师 程明明 教授

申请学位 工学硕士 培养单位 计算机学院

学科专业 计算机科学与技术 研究方向 计算机视觉

答辩委员会主席 邬霞 评 阅 人 匿名评审

南开大学研究生院

二〇二三年六月

南开大学学位论文使用授权书

本人完全了解《南开大学关于研究生学位论文收藏和利用管理办法》关于南开大学 (简称“学校”) 研究生学位论文收藏和利用的管理规定，同意向南开大学提交本人的学位论文电子版及相应的纸质本。

本人了解南开大学拥有在《中华人民共和国著作权法》规定范围内的学位论文使用权，同意在以下几方面向学校授权。即：

1. 学校将学位论文编入《南开大学博硕士学位论文全文数据库》，并作为资料在学校图书馆等场所提供阅览，在校园网上提供论文目录检索、文摘及前 16 页的浏览等信息服务；

2. 学校可以采用影印、缩印或其他复制手段保存学位论文；学校根据规定向教育部指定的收藏和存档单位提交学位论文；

3. 非公开学位论文在解密后的使用权同公开论文。

本人承诺：本人的学位论文是在南开大学学习期间创作完成的作品，并已通过论文答辩；提交的学位论文电子版与纸质本论文的内容一致，如因不同造成不良后果由本人自负。

本人签署本授权书一份（此授权书为论文中一页），交图书馆留存。

学位论文作者暨授权人 (亲笔) 签字：_____

20　　年　　月　　日

南开大学研究生学位论文作者信息

论 文 题 目	基于卷积注意力机制的视觉识别和语义分割骨干网络设计				
姓　　名	陆承泽	学号	2120200430	答辩日期	2023 年 5 月 15 日
论 文 类 别	博士 ☐　学历硕士 ☑　专业学位硕士 ☐　同等学力硕士 ☐　划☑选择				
学院（单位）	计算机学院		学科/专业 (专业学位) 名称		计算机科学与技术
联 系 电 话	13585230355		电子邮箱		czlu919@outlook.com
通讯地址 (邮编)：天津市海河教育园区同砚路 38 号南开大学津南校区计算机学院（300350）					
非公开论文编号			备注		

注：本授权书适用我校授予的所有博士、硕士的学位论文。如已批准为非公开学位论文，须向图书馆提供批准通过的《南开大学研究生申请非公开学位论文审批表》复印件和“非公开学位论文标注说明”页原件。

南开大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下进行研究工作所取得的研究成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：_____ 年 月 日

非公开学位论文标注说明

(本页表中填写内容须打印)

根据南开大学有关规定，非公开学位论文须经指导教师同意、作者本人申请和相关部门批准方能标注。未经批准的均为公开学位论文，公开学位论文本说明为空白。

论文题目			
申请密级	☐ 限制 (⩽2 年) ☐ 秘密 (⩽10 年) ☐ 机密 (⩽20 年)		
保密期限	20 年 月 日至 20 年 月 日		
审批表编号		**批准日期**	20 年 月 日

南开大学学位评定委员会办公室盖章 (有效)

注：限制 ★2 年 (可少于 2 年); 秘密 ★10 年 (可少于 10 年); 机密 ★20 年 (可少于 20 年)

摘要

作为计算机视觉领域中基础任务之一，视觉骨干网络设计任务旨在提取高语义、多尺度的视觉特征。先前的骨干网络大多由纯卷积模块或自注意力机制构建，针对卷积神经网络无法捕捉全局信息和基于自注意力的 Transformer 网络训练成本较高的问题，本文主要研究使用卷积注意力机制来简化自注意力机制，使用卷积注意力机制作为基础模块搭建骨干网络，并在多个下游任务的基准数据集上进行骨干网络的评估。对于特定的下游任务语义分割，本文基于卷积注意力提取多尺度信息，并在多个语义分割数据集进行评估。本文的主要贡献可概括如下：

（1）针对自注意力机制的计算复杂度高达 $\mathcal{O}(n^2)$ 的问题，本文提出了卷积注意力机制。使用 Hadamard 积作为融合方式，对输入特征进行重新加权，在降低计算复杂度的同时保证了对输入的自适应性。利用卷积注意力机制，本文从多个维度精心设计了 Transforemr 风格的骨干网络，包括模型深度与宽度的权衡，并且探索使用更大卷积核时骨干网络的性能变化。本文在三个不同的基准任务上进行了广泛的实验，包括图像识别、目标检测和语义分割。实验结果表明，本文提出的骨干网络在三个基准数据集上超越了当前基于卷积的和基于 Transformer 的最先进的方法。此外，本文还指出了该骨干网络存在的局限性和未来研究方向。

（2）针对特定的下游任务语义分割，本文针对性地拓展了所提出的卷积注意力机制，并简化了前有语义分割网络的设计理念，在减少计算量的同时避免性能的退化。考虑到多尺度特征的提取在语义分割任务中的重要作用，本文使用 Hadamard 积作为融合方式，在融合权重中利用不同大小的卷积核以引入多尺度信息。同时，本文使用大核卷积以扩大视觉感受野，有效提高了模型表现。本文在五个语义分割数据集以及一个遥感分割数据集上对本文提出的语义分割骨干网络进行了评估与实验结果分析。实验结果表明，相较于现有的语义分割模型，本文所提出的骨干网络在计算量相当的情况下性能大幅领先。

关键词： 视觉骨干网络；图像识别；语义分割；目标检测

Abstract

The design of vision backbone networks is one of the fundamental tasks in the field of computer vision, aimed at extracting high semantic, multi-scale visual features. Previously, vision backbones were mostly constructed using either pure convolutions or self-attention mechanisms. However, convolutional neural networks cannot capture global information due to their inherent locality, and the Transformer network based on self-attention has a high computational cost. This paper mainly studies the design and use of convolutional attention mechanisms to simplify self-attention mechanisms, using convolutional attention mechanisms as the basic building blocks to construct vision backbones, and evaluating the performance of backbone networks on multiple downstream benchmark datasets. For the specific downstream task of semantic segmentation, this paper slightly modifies the convolutional attention mechanism to extract multi-scale information and evaluates it on multiple semantic segmentation datasets. The main contributions of this paper can be summarized as follows:

(1) Design a convolutional attention mechanism to overcome the drawbacks of self-attention mechanisms. The convolutional attention mechanism uses Hadamard product as the fusion way to re-weight the input features, which introduces adaptive features while avoiding the computational complexity of $\mathcal{O}(n^2)$ in self-attention. Using the convolutional attention mechanism, this paper carefully designs Transformer-style backbone networks from multiple dimensions, including the trade-offs between model depth and width, and explores how performance changes when using larger convolutional kernels. This paper conducts extensive experiments on three different benchmark tasks, including image recognition, object detection, and semantic segmentation. The experimental results show that the proposed backbone network outperforms the current state-of-the-art methods convolution-based or Transformer-based networks on all three benchmark datasets. Finally, the limitations of the proposed backbone network are left for future research.

(2) When facing the specific downstream task of semantic segmentation, in order to

further explore the potential of the convolutional attention mechanism-based backbone networks, this paper slightly modifies the convolutional attention mechanism for better adaptation. Since the extracting multi-scale features is particularly important in semantic segmentation tasks, this paper also uses Hadamard product as the fusion method, while introduces multi-scale information by using different sizes of convolutional kernels as the fusion weights. At the same time, this paper uses larger-kernel convolution to expand the visual receptive field and improve the model's performance. In the macro-design of the model, this paper simplifies the design concept of the previous semantic segmentation networks, which can reduce computational complexity while avoiding performance degradation. This paper evaluates the proposed semantic segmentation backbone network on five semantic segmentation datasets and one remote sensing segmentation dataset, and analyzes the experimental results. The experimental results show that the proposed backbone network achieves a significant performance improvement over previous semantic segmentation models under similar computational complexity.

Key Words: Vision Backbone; Image Recognition; Semantic Segmentation; Object Detection

目录

第一章 绪论

第一节 研究背景与意义

随着深度学习技术的迅速发展，人类社会正逐渐步入一个全新的智能化时代。从 AlexNet[1]到 MAE[2]再到 ChatGPT，深度学习相关技术引领着各个领域的革命性变化。例如，在街道上的自动驾驶小车和校园中的无人配送快递车等等，都是深度学习技术在实践中的具体应用。在计算机视觉领域，传统的手工设计特征，例如加速稳健特征（SURF，Speeded Up Robust Features）、尺度不变特征变换（SIFT，Scale-Invariant Features Transform）、方向梯度直方图（HOG，Histogram of Oriented Gradient）等，早已被卷积神经网络（CNNs，Convolutional Neural Networks）取代。早期模型以卷积为构建基础来处理图像数据，典型工作例如 ResNet[3]、DenseNet[4]等等。但是传统的卷积神经网络存在一定的局限性，如无法对图像特征进行全局建模和无法很好地适配不同尺寸的输入图像。Vision Transformer（ViT）[5]的提出将计算机视觉领域推向了一个新的纪元，在图像分类、目标检测和图像分割等任务中均表现出极佳的性能。数据驱动的模型已然成为各个方向的研究主流，引起了广泛的关注。

目前热门的 ViTs 中的核心模块是自注意力模块。“注意” 源于人类认知过程中对海量数据的自动过滤的机制，属于认知过程的一部分。仿照这一机制，计算机视觉中的注意力机制自适应地选择图像中的重点区域。对于自注意力机制而言，其通过线性映射层对输入特征进行计算相似度矩阵并且利用矩阵乘法得到最终输出，相较于传统的卷积操作，自注意力能够对全局的特征进行建模，捕捉长距离依赖关系，并且其矩阵运算经过优化后在图形处理单元上运行效率较高。然而，基于自注意力机制的视觉 Transformer 通常需要更加高昂的训练代价，主要概括为数据代价与算力代价。首先，训练视觉 Transformer 需要更大规模的数据量，例如，当 ViT 首次被提出时，其需要使用 300M 的内部数据进行训练才能够得到很好的收敛效果。虽然后续例如 Swin-Transformer[6]的一些工作通过引入类似卷积的局部注意力能够加快模型的收敛速度，但是若希望得到一个性能好、泛化能力强的大模型，数以百万计的训练数据是必不可少的。其次，训练视

觉 Transformer 的算力代价也是不可忽视的。ViT 原文中表明，训练一个 ViT-H 模型（约 650M 参数量）若使用一个 TPUv3 核心，则需要约 2500 天的训练时间，这样的算力代价是难以负担的。

继 ViT 被提出，类 ViT 式的架构被运用在各种骨干网络中，例如 MLP-Mixer[7]模仿 ViT 的整体架构，在每个构建模块中仅仅使用多层感知机也取得了令人印象深刻的性能。后续工作 MetaFormer[8]指出，ViT 的成功在于其整体的架构，与其中的构建模块并无太大的关系，简单地使用池化操作在整体的架构下同样也能够取得很不错的效果。注意力机制的成功源于其强大的全局建模能力，而与注意力机制类似，使用大核卷积也能对长距离依赖关系进行建模。基于此，ConvNeXt[9]工作中探索了使用更大核的卷积构建一个强大的骨干网络。然而，该工作更加关注宏观与微观的网络设计，并且其发现当卷积核大于 7×7 时性能趋向于饱和。

不同于上述工作，本文的想法更加直接，旨在使用大核卷积设计卷积注意力模块以替换自注意力模块。本文从大核卷积出发设计卷积注意力机制，发现将卷积核提高至 21×21 时，网络性能才会趋于饱和，打破了 ConvNeXt 中对卷积核大小的限制。本文在视觉 Transformer 流行的当下，利用大核卷积完成类似注意力操作，以此为构建模块搭建新的层级式视觉骨干网络，在多个基准数据集，包括 ImageNet-1k、Microsoft COCO 以及 ADE20K，取得了非常不错的效果。具体来说，本文第三章提出的 Conv2Former 网络，在没有使用 ImageNet-22k 数据集预训练的情况下，在 ImageNet-1k 数据集上以 90M 的网络参数量取得了 84.4% 的 Top-1 准确率，在类似的参数量情况下相较于 ConvNeXt[9]提高了 0.6%。此外，利用设计的卷积注意力机制，本文为语义分割任务精心定制了一个骨干网络，基于此骨干网络搭建语义分割模型并在多个语义分割的基准数据集上取得了最先进的性能。具体地，本文第四章提出的 SegNeXt 网络，在类似的参数量情况下相较于前有的最先进的方法 Mask2Former[10]提高了 3.3%，仅仅使用接近 49M 的参数量取得了 51.0% 的 mIoU。

第二节 国内外研究现状

本文主要研究卷积注意力的设计，本节将介绍卷积神经网络以及最近的 Transformer 架构的研究进展，对现有的基于注意力机制的方法进行概述，最后介绍应用的任务场景，即语义分割任务的相关进展。

1.2.1 卷积神经网络

自卷积神经网络时代以来，视觉识别模型的成功主要归功于其发展，例如AlexNet[1]，VGGNet[11]和 GoogLeNet[12]。AlexNet[1]率先使用卷积、ReLU 激活、池化等模块搭建了一个非常简单的卷积神经网络并在 ImageNet 分类比赛中取得了非常优异的成绩。随后 VGGNet 与 GoogLeNet 搭建了略微更深一些的网络并取得了更高的性能。上述这些模型虽然只包含了不到 20 层的卷积，但是梯度消失的问题已然存在。ResNet[3]通过引入残差连接推动了传统的卷积神经网络的发展，使训练非常深的模型成为可能。Inception 系列工作[13-14]和 ResNeXt[15]进一步丰富了卷积神经网络的设计原则，并提出使用具有多个并行链路的、以及专门设计的卷积过滤器的构建模块。

SENet[16]及其后续一些工作[17-18]并没有继续调整网络架构，而是旨在通过轻量级注意力模块来改进卷积神经网络，这些模块可以显式地对通道之间的相依赖性进行建模。随着神经网络架构搜索方向的发展，EfficientNet 系列[19-20]和MobileNetV3[21]利用该技术[22]来搜索高效并且性能极好的网络架构。此外，还有一些工作利用不同的训练或优化方法或微调技术[23-26]来推进 EfficientNet 的发展。RegNet[27]探索了如何去设计卷积神经网络的设计空间，并且阐述了一些网络设计的通用准则。直到最近，NFNet[26]通过设计了一个无归一化的网络架构终于击败了 EfficientNet 的性能，这是第一个没有使用额外数据、在 ImageNet-1k 数据集上达到 86.5% 的 Top-1 准确率的网络。

多年来，卷积神经网络作为视觉识别领域的主流网络确实非常成功，它们的重点是如何通过设计更好的架构来学习更具辨别力的局部特征，但是在[28]中已被证明，从本质上来看卷积神经网络是不能显式地建立全局关系的，而该能力在网络建模中十分重要。

近年来，一些工作[9,29-32]旨在使用具有大卷积核的卷积来对全局的信息进行建模，简单直接地扩大局部卷积的感受野，并且学习了 Transformer 网络的整体架构优势，掀起了一波新的卷积神经网络的浪潮。其中一个典型的例子是VAN[31]，它利用标准的深度卷积和扩张卷积来对大卷积核进行分解。与 VAN 不同，本文第三章中的 Conv2Former 并不旨在分解大卷积核，而是展示将自注意力机制替换为卷积注意力机制，这也可以使网络获得良好的识别性能。

1.2.2 注意力机制

人类的注意力机制可以简要描述为从大量信息中利用有限的注意力筛选出最有价值的信息。在深度学习时代，注意力机制也同样借鉴于人类的感知方式，在自然语言处理领域中被首次提出并取得了巨大的成功。简单来说，注意力机制是一种自适应选择的过程，旨在使得网络能够关注于特征中重要的部分。总的来说，在视觉领域中最常见的注意力机制是通道注意力和空间注意力，不同类型的注意力机制在网络中扮演着不同的角色。例如，空间注意力主要关注于重要的空间区域[5-6,33-35]，典型的是自注意力机制，首先使用多个线性映射层生成查询特征、键特征以及值特征，通过使用查询特征与键特征生成相似度矩阵，该矩阵为空间中重要的区域分配更大的权重，随后与值特征进行矩阵乘法得到结果；而先前一些工作[16,36-37]表明，通道注意力更关注于使网络有选择的去关注图片中重要的物体。SENet[16]在卷积神经网络中率先提出通道注意力，即 Squeeze-and-Excitation 模块，使用多个线性映射层计算得到特征图每个通道上的权重，随后使用该权重对特征图的每个通道进行重新加权。空间注意力在视觉领域率先由 Wang 等人[28]提出，其使用非局部模块，计算特征图中两个位置的相似度得到空间相似度矩阵，完成注意力操作。后续工作中最常使用的是自注意力操作，其本是在自然语言处理任务中提出，但是随着 Vision Transformer（ViT）[5]在视觉领域的走红，自注意力操作已然成为了视觉领域的主流构建模块。值得注意的是，由于自注意力机制需要使用多次矩阵乘法，其计算复杂度与输入序列的长度构成平方关系。

1.2.3 视觉 Transformer

起源于自然语言处理任务的 Transformer 现如今已成功应用于视觉领域。其中第一个成功将 Transformer 迁移到视觉领域中的工作是 ViT[5]，其展示了 Transformer 在处理图像分类问题中大规模数据的巨大潜力。随后，DeiT[38]通过使用强数据增广方法以及知识蒸馏方法，使得网络在没有大规模数据集的情况下也能够进行良好的训练。受卷积神经网络中金字塔架构的成功启发，一些工作[6,39-43]为了利用多尺度的特征，设计了金字塔结构的视觉 Transformer 进一步提高了性能。例如，PVT[40]工作对自注意力机制进行优化，在自注意力过程中加入对序列长度降采样的操作以减少计算量，并且引入了层级式的网络结构，对于下游任务更加友好。Swin-Transformer[6]工作以窗口为基础，在窗口内完成自

注意力机制，在减少计算量的同时引入了局部信息，与此同时使用窗口滑动操作完成全局信息的交互，在 Transformer 的基础上将全局信息与局部信息进行融合，取得了当时最优异的性能。在 Swin-Transformer 之后，一系列工作[44-52]仿照窗口注意力机制的操作对注意力进行重新设计，引入局部依赖性，达到新的最先进性能。除此以外，同样有一些工作[53-58]探索了 ViT 在图像识别领域的可扩展能力，即当数据规模或模型规模扩大时，整个框架能够取得的性能是否能够提升的能力。具体来说，Yuan 等人[57]首次展示了使用两阶段的 ViT 在 ImageNet 上能够超越当时最先进的卷积神经网络。随后，基于 ViT 的一些自监督学习策略[2,59]也开始涌现，利用大量无标签的数据来提高普通视觉 Transformer 的泛化能力。与上述工作不同，本文第三章中的 Conv2Former 的工作旨在设计一种新的注意力机制和架构，以直接提高基于 Transformer 架构的卷积神经网络的能力，而无需预训练和额外的训练数据。

1.2.4 语义分割

语义分割是计算机视觉中的根基任务。自从 FCN[60]被提出，卷积神经网络（CNNs）[61-69]就已逐渐成为语义分割的主流网络架构并取得了巨大的成功。FCN[60]完全使用卷积神经网络搭建，将语义分割任务视为逐像素分类任务，完全超越传统图像分割方法。随后，著名的 DeepLab 系列工作[70]使用扩张卷积提高网络的感受野，多分支的结构帮助模型提取多尺度的特征。近年来，随着 Transformer 的提出与发展，基于 Transformer 的语义分割方法[10,71-77]相较于基于卷积神经网络的方法已经展示了其巨大的潜能。在深度学习年代，大部分分割模型的架构大致可以划分为两个部分：编码器和解码器。对于编码器，研究者通常采用当下流行分类网络（例如 ResNet[3]、ResNeXt[15] 和 DenseNet[4]），然而，语义分割作为一个密集预测任务，与图像分类任务本身是截然不同的，图像分类任务中模型仅仅需要捕捉到图像的语义类别，并不关系图像中的每一个细节。并且，目前 ImageNet-1k 数据集中的图片大多是单物体且以物体为中心，与实际的自然图片相差巨大，大大降低了图像分类的难度，并不适配于语义分割任务。因此在分类任务中获得的性能提升可能并不会为语义分割任务带来任何提升[78]。考虑到上述的问题，一些精心设计的编码器随之出现，包括 Res2Net[68]、HRNet[67]、SETR[71]、SegFormer[72]、HRFormer[73]、MPViT[79]和 DPT[75]等等。具体来说，Res2Net[68]修改了 ResNet 中的瓶颈模块，在瓶颈模块中划分多个尺度旨在捕获图像中更细粒度的特征。而 HRNet[67]的做法更加简单直接，由于网络

中的下采样过程会造成信息的部分丢失，它抛弃了对图像不断进行下采样的过程，将网络中的特征图保持在一个较高的分辨率下进行运算，并且不断与低分辨率的特征融合进行信息的交互，完成对图像的建模。SETR[71]是第一个将 ViT 架构引入到语义分割任务的工作，其整体保持编码器-解码器架构，在编码器中使用 ViT 的架构风格，利用自注意力机制强大的编码能力在多个语义分割数据集上取得了与卷积神经网络相当的效果。SegFormer[72]同样优化语义分割任务中的编码器部分，模仿 PVT 工作[40]对自注意力机制的复杂度进行优化，同时使用更小的块大小，对像素级分类任务更加友好。HRFormer[73]延续 HRNet 的设计理念，在高分辨率特征图中不断融合低分辨率特征，利用注意力机制大大提高了模型的性能。MPViT[79]使用 ViT 的总体架构，在块嵌入部分便将输入图像进行多个尺度的编码，不同尺度编码的 token 送入到同样的 ViT 架构中，在网络的输出部分再对多个尺度的图像 token 进行信息的聚合完成多尺度信息的融合。

对于语义分割任务中的解码器，其通常被用于配合编码器来获得更好的分割性能。不同类型的解码器能够帮助编码器取得不同的目标，例如：为了获得多尺度的感受野，Zhao 等人[64]使用一个类金字塔架构的解码器对多尺度的特征进一步精细化。经典的 U-Net[62]解码器对编码器中的多尺度语义信息进行一步步的收集并上采样得到最终的分割结果。除此以外，一些工作[80-84]在分割工作中引入其他先验信息来增强分割效果，典型工作是 Zhen[80]等人在解码器中提出了 PCM 模块提取多尺度信息并且利用边缘信息来增强分割效果。最近的一些基于注意力机制[65,85-90]的解码器，利用不同形式的注意力机制捕捉全局上下文信息。

第三节　主要研究内容

骨干网络是计算机视觉领域中最核心的研究内容，如何设计骨干网络是长期的研究热点。现有工作一般基于卷积或注意力机制搭建骨干网络，针对此类方法存在的问题与挑战，本文提出了以卷积注意力机制为核心的骨干网络，并在下游任务中成功应用，具体思路如图 1.1所示。

视觉领域目前主流的骨干网络多数基于卷积或注意力机制，存在一定的性能瓶颈。以卷积为构建模块搭建的骨干网络[3-4,91]，因卷积具有局部性和静态性的天然特点，性能普遍低于后续的 Vision Transformer[5]。而 Vision Transformer 训练过程收敛困难且训练开销较高，难以满足具体场景中的效率要求。

针对上述两种结构存在的不足，本文对自注意力机制进行简化，提出了卷

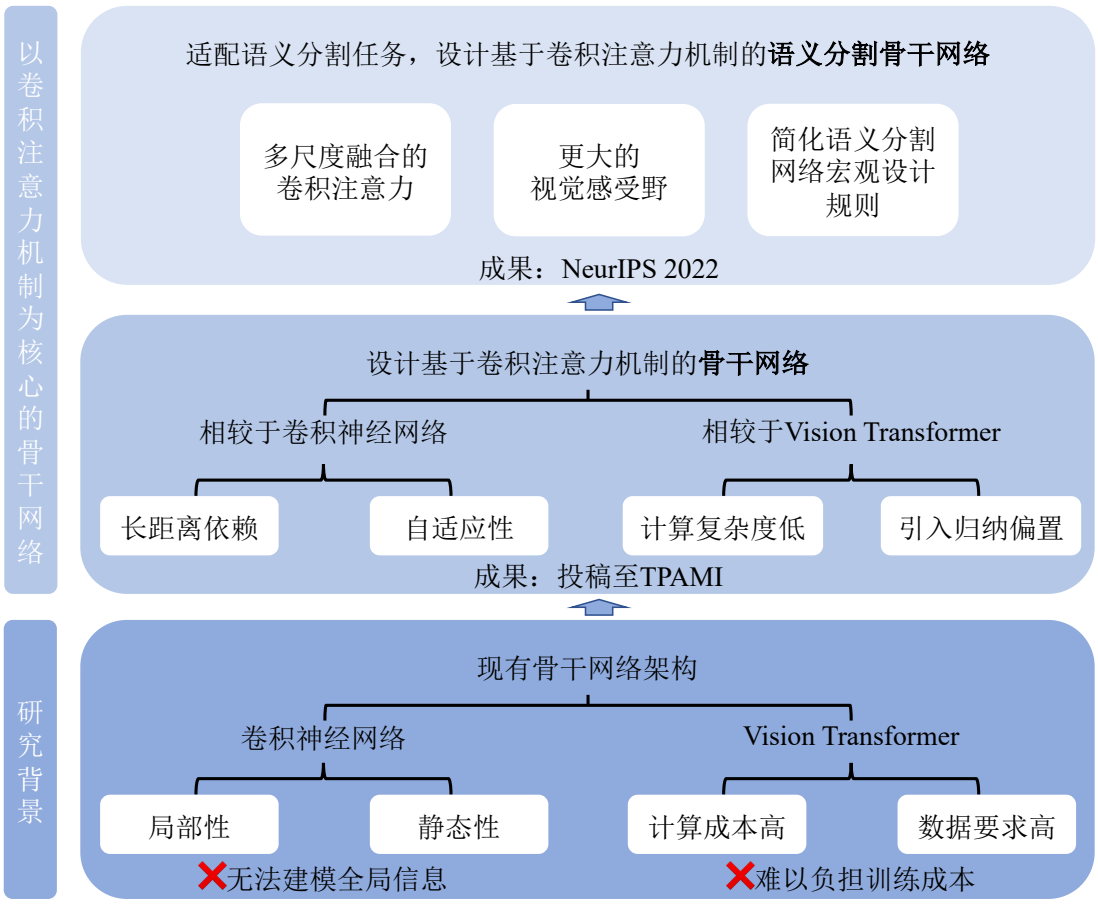


图 1.1 本文主要的研究内容。

积注意力机制，并以此为构建模块完成了骨干网络的设计。在自注意力机制的基础上，卷积注意力机制以 Hadamard 积替代矩阵乘法，有效减少了计算量，且卷积注意力的权重值由大核深度卷积生成，加快模型收敛速度。基于卷积注意力机制，本文探索了更大核的卷积（$\geq 7 \times 7$）对模型性能的影响，在多个数据集上（包括 ImageNet-1k、COCO 和 ADE20K）对该骨干网络的性能进行了评估。此外，本文在更大的数据集（ImageNet-22k）上进行预训练，并且同样在上述的一些数据集上对本文的模型进行了评估。相较于传统的卷积神经网络，本文所提出的骨干网络拥有更大的感受野，并且由于卷积注意力机制的优势，本文的模型与基于自注意力的模型一样拥有一定的自适应性，因此本文的模型能够取得更高的性能。相较于现有的 Vision Transformer，本文的卷积注意力机制计算复杂度更低，更加适用于实时任务的要求。由于卷积的特殊性，本文所提出的骨干网络无需大规模的预训练数据集即可快速地收敛，一定程度上克服了 Vision

Transformer 训练难度大的缺点。

本文还进一步探索了在特定任务场景（语义分割）下如何对卷积注意力机制进行适配使得充分发挥其潜力。不同于图像分类任务，语义分割任务本身为像素级分类任务，利用多尺度的特征对于语义分割任务来说已被证明是非常有效的[70]。为了进一步提高以卷积注意力为构建模块的骨干网络在语义分割任务上的性能表现，本文探索了如何构建一个能够捕获多尺度特征、动态的语义分割骨干网络。具体来说，在本文的构建模块中将会包含多个网络分支，对于每个网络分支，其包含不同核大小的深度卷积以构建多尺度特征。同时，使用深度卷积同样能够使得本文的方法更加轻量，甚至能达到实时推理的要求。多尺度的融合、强大的骨干网络以及对输入自适应调整的注意力权重使得本文专为语义分割任务设计的骨干网络拥有强大的性能。在多个著名的语义分割基准数据集上，包括 ADE20K[92]、Cityscapes[93]、COCO Stuff[94]等等，本文验证了所提出的语义分割网络的有效性。

第四节　全文组织结构

本文主要探索了利用卷积注意力机制设计的骨干网络在各个基准数据集上的性能，以及其在下游语义分割任务上的应用。本文共分为五个章节，每章内容具体安排如下：

本文的第一章为绪论，介绍了本文的研究背景以及意义，对国内外骨干网络和语义分割领域的研究现状进行了简要的概述，对本文的研究内容进行了简要的概括以及描述了章节安排。

本文的第二章详细介绍了与本文最相关的两项技术，包括 Vision Transformer[5]以及 ConvdNeXt[9]。

本文的第三章介绍了提出的卷积注意力设计理念，描述如何以卷积注意力为基础搭建一个强有力的骨干网络，并且对于该骨干网络给出了在各种基准数据集上的评价结果和对结果的分析。

本文的第四章首先分析了前有的成功的语义分割网络应当具备的属性，并对所提出的卷积注意力机制略微修改以适应语义分割任务，最终在语义分割的各种基准数据集上进行性能的评估并对结果进行了分析。

本文的第五章是总结和展望，对本文的内容进行了简要的总结并对未来工作进行了展望。

第二章 相关技术介绍

自从 AlexNet[1]的成功，卷积神经网络架构在图像分类和语义分割的研究中发挥了主要作用。这些任务的发展主要归功于骨干网络架构的不断提出，如 VGGNet[11]、ResNet[3] 和 DenseNet[4] 等等。自从 Vision Transformer（ViT）[5]的出现以来，它因其强大的表征能力和对不同输入模式的适应能力，迅速成为视觉领域的新明星。ViT 使用 Transformer 架构，将自然语言处理中广泛使用的自注意力机制引入了图像分类领域，以提高其性能，取得了相当大的成功。除了 ViT 之外，ConvNeXt[9]也是当前视觉领域中备受关注的技术之一。ConvNeXt 使用层级式结构，旨在提高模型的表征能力，同时兼顾模型的计算效率。本章将对与本文最相关的两个技术进行介绍，包括 ViT[5]以及 ConvNeXt[9]。

第一节 Vision Transformer

Vision Transformer[5]是一种利用自注意力机制来完成各类视觉任务的深度学习模型。它受启发于自然语言处理领域，并且在该领域其已被证明在处理序列数据方面具有极高的性能。而在视觉领域，Vision Transformer 模型在 2020 年由 Google Brain 团队提出，是一种全新的深度学习模型，旨在将图像建模为序列模型，利用自注意力机制来处理图像。

早些年，卷积神经网络（CNN）在各类视觉任务榜单上名列前茅，是最常用深度学习模型之一。但是，由于卷积本身的局部特性，卷积神经网络在处理图像中的长距离依赖性方面效果不佳，即使其通过更深层次的网络架构能够获得更大的感受野，也无法直接对全局的信息进行建模，在语义分割或目标检测等领域的性能也因此受到极大的限制。相比之下，Vision Transformer 使用自注意力机制来允许不同部分之间的信息交互，对图像中的上下文信息能够很好的建模，可以更好地捕捉图像中的长期依赖关系，这在传统的卷积神经网络中是不容易实现的。

Vision Transformer 中的构建模块如图 2.1所示。为了构建序列形式的输入，ViT 首先对输入图像进行 “序列化”，即将图像进行切割变成图像块，随后使用线性映射层将图像块映射为图像 token。由于自注意力模块对 token 的位置信息

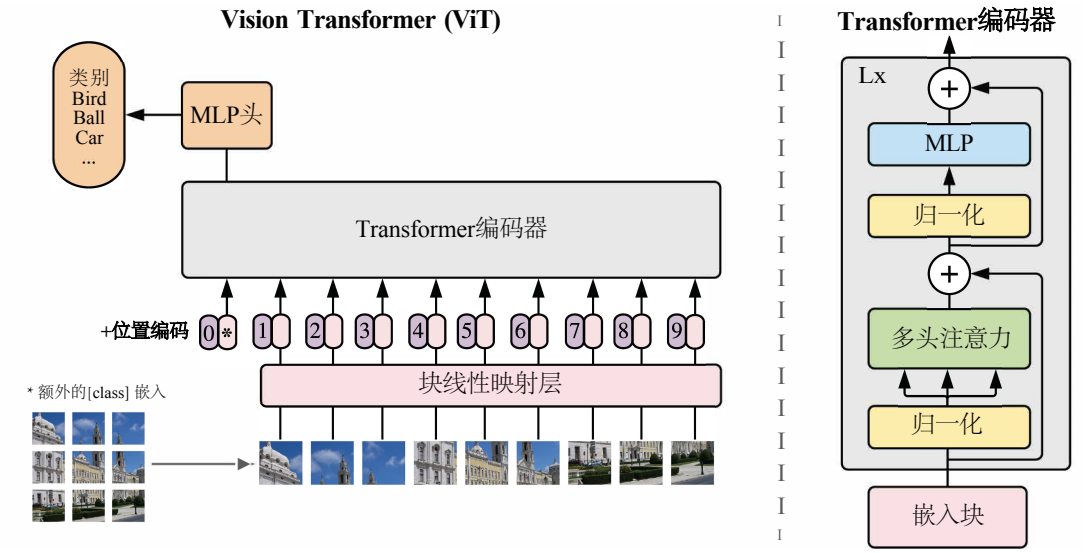


图 2.1 ViT 的总体架构。输入图片首先被切分为图片块，经过线性映射层后转变为 token，在送入 Transformer 编码器前，还需添加额外的 “类别”token 并且加上位置编码，经过多个自注意力模块后送入 MLP 头得到最终输出。图片来源于[5]。

并不敏感，需要构建与 token 序列长度相同的位置编码（Positional Embedding），一般使用 sin 位置编码形式或 cos 位置编码形式，通过与 token 进行加法完成对位置信息的捕捉。除此以外，一般会额外添加 “类别”token（Class Token），在该 token 后通常会添加上线性映射层作为分类头用于最终的分类任务。

整个 ViT 架构的核心是自注意力模块，自注意力模块允许模型对输入图片中的不同部分进行相对重要性的评判，并对其进行自适应的加权。这意味着模型可以根据输入中的不同部分进行精细的选择和组合，从而更好地捕捉图像中的结构和特征。同时，自注意力机制允许模型根据输入 token 中的不同部分学习不同的表征，在面临复杂的下游任务场景时拥有强大的表征能力，从而提高了模型的泛化性能。

具体来说，给定输入长度为 N 的图像块序列，首先通过线性映射层将其映射为 token 序列，即

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D}, \quad (2.1)$$

其中，$\mathbf{E}$ 为线性映射层，旨在将输入的图片块映射为 token 序列，$\mathbf{x}_{\text{class}}$ 为额外添加的 “类别”token。随后，假设 ViT 中共包含 L 个构建模块，则第 l 个构建模块

可定义为如下形式：

$$\mathbf{z}'_l = MSA(LayerNorm(\mathbf{z}_{l-1})) + \mathbf{z}_{l-1}, \qquad l = 1 \ldots L \tag{2.2}$$

$$\mathbf{z}_l = MLP(LayerNorm(\mathbf{z}'_l)) + \mathbf{z}'_l, \qquad l = 1 \ldots L \tag{2.3}$$

其中，MSA 为多头自注意力机制，是整个 ViT 的核心模块，MLP 为多层感知机，即由多个线性映射层组成。上述公式的图示表达如图 2.1右侧所示。对于自注意力机制 MSA，可简单表达为下式：

$$MSA(Q,K,V) = SoftMax(QK^T/\sqrt{d})V, \tag{2.4}$$

其中，Q、K、V 为对 token 序列进行线性映射后的查询、键、值特征。自注意力通过 Q 与 K 矩阵乘法生成相似度矩阵，最终同样使用矩阵乘法对 V 特征进行重新加权，以此获得自适应能力。而在具体实现中，为了实现 “多头” 的特性，会将通道 c 等分，随后在每个头上进行各自的自注意力操作而不互相干扰。构建模块中另一个比较重要的部分是多层感知机（MLP），其一般使用线性映射层与非线性激活层构成，旨在对不同 token 的相同通道上进行信息的交互。经过 L 个上述的构建模块后，能够得到最终输出 y，可用如下公式表示：

$$\mathbf{y} = LayerNorm(\mathbf{z}_L^0). \tag{2.5}$$

总之，Vision Transformer 是一种利用多头自注意力机制来处理图像的新型神经网络模型。相较于传统的卷积神经网络，ViT 没有引入与输入图像相关的归纳偏置，能够更好地捕捉图像中的长距离依赖关系，由于其强大的建模能力以及对输入图像的自适应能力，ViT 具有更优秀的泛化性能。另外，ViT 的提出使得基于 Transformer 的模型能够处理各种不同模态的数据，包括图片、视频、文本、语音等等。通过将各个模态归一到同种模型下，使得后续一些多模态大模型的工作成为了可能，例如著名的 CLIP[95]。CLIP 利用了 ViT 的强大表征能力，成功地将图片和文本联系起来进行学习，取得了在多个视觉任务上最先进的性能表现。除了图像任务外，ViT 还被应用于其他任务，如自然语言处理和语音识别等。这些应用进一步证明了 ViT 的多模态能力和强大表征能力，ViT 将在更多的领域中发挥越来越重要的作用。

第二节 ConvNeXt

自 ViT 问世以来，许多类似 ViT 的工作不断涌现，目的在于在 ViT 中引入归纳偏置或改进自注意力的复杂度。除了上述改进 ViT 的工作外，Tolstikhin 等人[7]在最近的研究中表明，单纯使用多层感知机构建骨干网络同样能够取得令人印象深刻的效果。为了克服当前卷积神经网络的局限性，Liu 等人[9]从目前的层级式 ViT 网络中吸取各种设计优点，在标准卷积神经网络的基础上，将其更改为类似于 Swin-Transformer[6]的层级式架构，称为 ConvNeXt。ConvNeXt 取得了优异的效果，超越了前有的标准卷积神经网络。相较于 ViT 模型，ConvNeXt 使用大卷积核，能够在引入归纳偏置的同时更快地收敛，且不需要进行大量的标注数据训练。接下来，本章将对 ConvNeXt 进行详细的介绍，描述其使用的设计方法，如图 2.2所示。

ConvNeXt-T 变体模型从 ResNet-50 网络[3]出发，经过宏观设计的改变、模仿 ResNeXt 的设计、使用逆瓶颈层、使用大卷积核以及微观设计的改变共五个阶段，在 ImageNet-1k 取得了最先进的结果，接下来将详细介绍具体的改变方式。

宏观设计。传统的 ResNet 以及其他一些卷积神经网络设计通常会将整个骨干网络划分为几个网络块，然后在每个块中对特征图进行降采样操作以减小特征图的分辨率，从而在提取高层级语义特征的同时实现减少计算量的目的。然而，在 Swin-Transformer 中，骨干网络被划分为多个阶段，每个阶段包含多个构建模块，每个构建模块由自注意力块和多层感知机组成。每个阶段输出的特征图具有不同的尺寸，从而产生多尺度的特征图。具体来说，Swin-Transformer 共划分为 4 个阶段，而各个阶段中构建模块的数量比例一般为 1 : 1 : 3 : 1。在 ConvNeXt 中，首先将 ResNet-50 的每个阶段的构建模块比例设置为 1 : 1 : 3 : 1，在 ImageNet-1k 数据集上可以实现 79.4% 的 Top-1 准确率。除此之外，ConvNeXt 还借鉴了 Swin-Transformer 中的设计，在模型的开始添加了 stem 层，旨在对输入图像进行降采样操作。这个策略能够在 79.4% 的基础上进一步提高 0.1% 的准确率。

ResNeXt。在许多轻量级网络[96-97]中，作者往往通过使用深度卷积来大量地减少模型计算量。这里，ConvNeXt 同样使用深度卷积对计算量与准确率进行一定的权衡。直接使用深度卷积同样也会带来一定的性能下降，当将卷积替换为深度卷积时，模型性能下降至 78.3%，但是模仿 ResNeXt 的设计思想，将网络

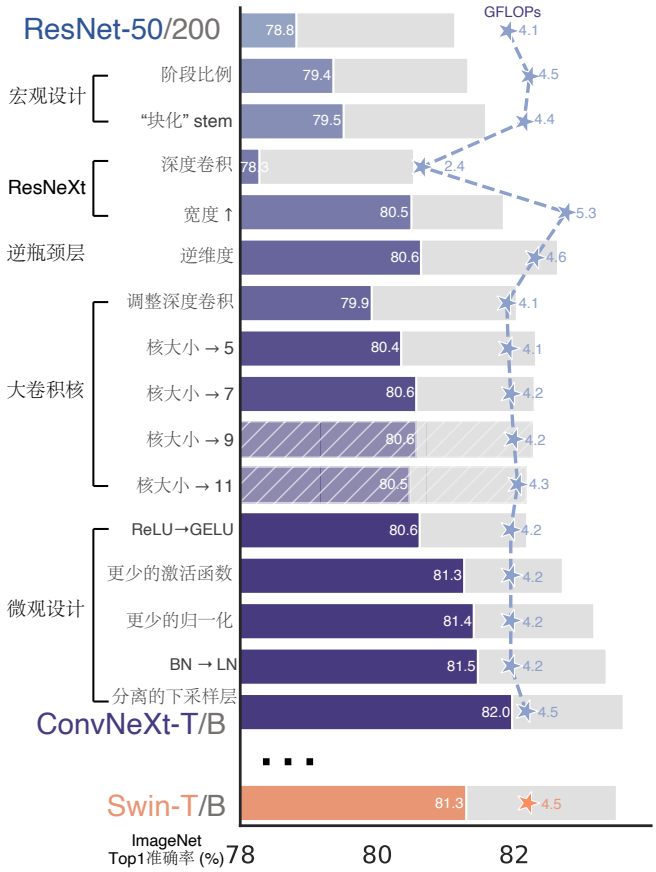


图 2.2 ConvNeXt 中将一个标准的卷积神经网络改变为层级式的 Transformer 架构所做出的调整。图片来源于[9]。

的宽度提高后，模型性能显著提升至 80.5%。

逆瓶颈层。瓶颈层早在 ResNet[3]中被提出，其是一个中间维度小、两头维度大的多层线性映射结构，如图 2.3（a）所示。而 MobileNet-v2 网络[97]使用了逆瓶颈层的架构，其中间维度大，两头维度小，这样能够很有效地避免信息的流失。ConvNeXt 同样也采用了逆瓶颈层的架构，如图 2.3（b）所示。使用逆瓶颈层能够在性能没有降低的情况下大幅度降低计算量。

大卷积核。接下来，ConvNeXt 探索了使用更大核的卷积。首先将逆瓶颈层中的深度卷积的位置进行调整，如图 2.3（c）所示。该操作能够将计算量进一步降低，与此同时性能也有所下降。调整位置后在 ImageNet-1k 的 Top-1 准确率仅有 79.9%。随后，ConvNeXt 尝试在逆瓶颈层中使用不同核大小的卷积核。由于使用两个 3×3 的卷积与使用一个 5×5 的卷积能够取得相同的感受野，并且使用两个 3×3 的卷积通常性能更优，因此在传统的卷积神经网络工作中通常使用

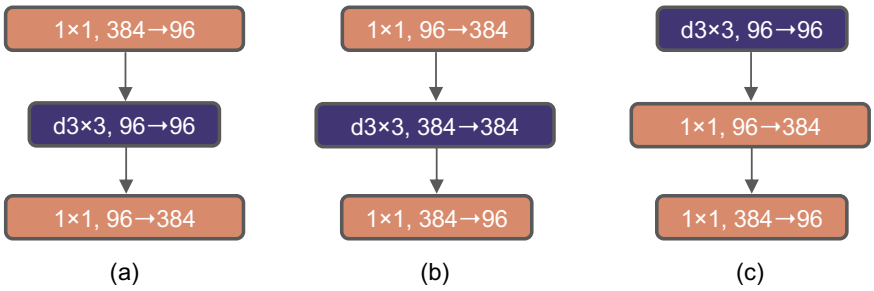


图 2.3 不同的瓶颈层的形式。（a）ResNeXt 中使用的瓶颈层；（b）MobileNet-v2 中使用的逆瓶颈层；（c）ConvNeXt 将深度卷积前移后的逆瓶颈层。图片来源于[9]。

3 × 3 卷积进行堆叠。而 ConvNeXt 尝试了不同核大小的深度卷积，包括 5 × 5、7 × 7、9 × 9 以及 11 × 11。作者发现使用更大核的深度卷积能够取得更优的效果，并且当卷积核的大于 7 × 7 时，在带来计算负担的同时不会带来更多的性能提升。因此 ConvNeXt 选用 7 × 7 的深度卷积。

微观设计。除了上述的一些改变外，ConvNeXt 对模型的微观设计方案包括多个方面的优化，不仅使用了 ViT 中最常见的 GELU 激活函数取代了卷积神经网络中通常使用的 ReLU 激活函数，而且针对卷积网络中的激活函数和归一化操作都进行了优化，使网络使用更少的激活函数和归一化，并将批归一化操作替换为层归一化操作以进行更好的模型优化。ConvNeXt 还对降采样层进行拆分等处理，使得模型参数感知性更强，进一步提高模型性能。这些设计灵感的大部分都来源于最近流行的 Swin-Transformer 工作[6]。值得一提的是，ConvNeXt 使用的下采样策略与传统的卷积神经网络不同。传统方法通常会使用 3 × 3、步长为 2 的卷积对特征图进行下采样，而这些下采样操作通常直接嵌入在网络中，但 ConvNeXt 将下采样操作布置于每个阶段之前，这个策略简单而有效，最终能够使得 ConvNeXt-T 的准确率达到了 82.0%，超过了 Swin-Transformer 并在 ImageNet-1k 数据集上达到最先进的结果。

通过上述这些方式，ConvNeXt 可以更细粒度地对特征进行建模，避免特征空间中的信息丢失和冗余。与此同时，ConvNeXt 仍然能够保持较高的计算效率，具有广泛的应用前景。ConvNeXt 已经被应用在图像分类、目标检测和分割等多个视觉任务中，取得了令人瞩目的结果。总的来说，ConvNeXt 的提出打破了自注意力在各类视觉任务上的统治地位，为卷积神经网络的设计提供了新的思路，并且将卷积神经网络的应用领域进一步扩展到更加复杂的视觉任务中去。随着对 ConvNeXt 的不断深入研究，相信它将为深度学习领域的发展注入新的活力。

第三节 本章小结

本章介绍了与本文的方法最相关的两个工作，即 ViT 与 ConvNeXt，这些工作对后续工作产生了深远的影响。对于 ViT 而言，其是第一个尝试将视觉信息的处理与自然语言的处理统一的工作。ViT 成功地将 Transformer 架构引入了图像分类领域，显著提高了图像分类的准确率，并为多模态数据处理提供了新的思路和可能性。ViT 的成功是建立在自注意力机制的基础之上，其在不同任务上的优秀表现也验证了自注意力机制在视觉任务中的适用性。对于 ConvNeXt，其对卷积神经网络进一步的探索，为卷积神经网络注入了新的活力。ConvNeXt 采用更大的卷积操作来提高模型的表示能力，在网络的宏观与微观设计方面平衡了建模能力与计算效率。ConvNeXt 的提出展示了卷积神经网络的另一种设计思路，对视觉任务的处理提供了新的解决方案。与这些工作不同的是，本文旨在使用纯卷积构建骨干网络，设计卷积注意力机制，并以此为基础将 ViT 的优势带入到卷积神经网络中，并成功将其应用于下游任务进行验证。本文提出的方法不仅融合了注意力机制的优势，而且在计算效率方面具有较高的优势。这一成果结合了各类神经网络的优势，并为卷积神经网络设计提供了新思路和途径。

第三章　基于卷积注意力机制的骨干网络设计

骨干网络是计算机视觉模型中的基础元素之一，旨在为一系列的下游任务诸如图像识别、目标检测、语义分割等等任务提供良好的视觉特征。近年来，骨干网络的设计一直是研究的热点，从早期基于卷积的 AlexNet[98]到基于自注意力机制的 Vision Transformer（ViT）系列网络[5]。目前，基于 Transformer 网络的性能已经逐渐领先于基于纯卷积的网络，并且已逐渐成为视觉领域的主流，但值得注意的是，纯粹的自注意力操作因包含多次矩阵乘法操作，其计算复杂度与输入序列长度成平方关系，训练代价通常较大。本章的研究目标，在于探索如何更高效地利用更大核的卷积来取代自注意力机制，捕捉长距离依赖关系，通过利用卷积注意力机制进一步提高骨干网络的性能。

具体而言，本章节首先阐述了本研究的动机和贡献，接着详细介绍了基于卷积自注意力机制的骨干网络设计。最后，本章在一些公开数据集上进行了实验，包括图像分类任务（ImageNet-1k 数据集[99]）、目标检测/实例分割任务（COCO 数据集[94]）、语义分割任务（ADE20K[92]），并对实验结果进行了分析和讨论。此外，本章讨论了该骨干网络的局限性留待未来研究。总之，本章节的主要贡献在于提出一种基于卷积注意力机制的骨干网络，旨在解决自注意力机制计算复杂度高的问题，并在三个公开数据集上取得了良好的表现。本章希望能够为骨干网络的进一步发展提供一些有价值的思路和方法。

第一节　研究动机以及贡献

2010 年代计算机视觉识别领域取得了巨大的进展，其中大部分归功于卷积神经网络，其中以 VGGNet[11]、Inception 系列[12-14]和 ResNet 系列[3,15,68,100]为代表等模型为主。这些图像识别模型为了获取更大的感受野，通常会堆叠多个构建模块，并且采用类似金字塔网络架构，但是这样的架构很难显式地建模全局上下文信息，而全局的上下文信息对于密集型预测的下游任务，例如目标检测、语义分割而言，十分重要。SENet 系列[16-18,101]打破了 CNN 的传统设计，引入了基于注意力机制的方法来捕捉长远程依赖关系，取得了惊人的性能表现。

从 2020 年开始，Vision Transformers (ViTs)[5-6,38,40,57] 进一步推动了视觉识别

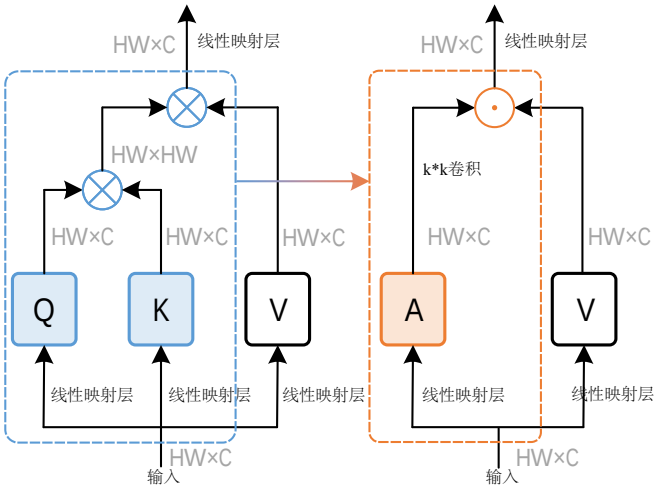


图 3.1 自注意机制和本章所提出的卷积注意力操作之间的比较。可以看出，本章并没有通过查询和键之间的矩阵乘法来生成注意力矩阵，而是直接使用 $k \times k$ 的深度卷积来生成权重，通过 Hadamard 积（⊙：Hadamard 积；⊗：矩阵乘法）对值特征进行重新放缩。

模型的发展，并在 ImageNet 分类和下游任务上展现出比当前最先进的卷积神经网络[19-20]更好的结果。这是因为相对于仅仅提供局部连接的卷积，Transformers 中的自注意力机制能够建模全局的成对的依赖关系，提供了一种更有效的方式来编码图片中的空间信息，正如[102]中所展示的一样。然而，当处理高分辨率图像时，自注意力机制因其计算机与 token 数量成平方级增长关系，也存在计算量过大这样的问题。

最近，一项名为 ConvNeXt[9]的有趣工作揭示了通过简单地对标准 ResNet 进行一步步改进，并使用与 Transformers 相似的设计和训练方法，最终得到的卷积神经网络甚至可以表现得比一些流行的 ViT 模型[6,40]还要好。RepLKNet[29]利用重参数化技术也展示了利用大卷积核进行视觉识别的潜力。这些探索鼓励许多研究者通过利用大卷积核[31,103]、高阶空间交互作用[30]、稀疏卷积核[32]等方式重新思考卷积神经网络的设计。到目前为止，如何更有效地利用卷积构建强大的卷积神经网络架构仍然是计算机视觉领域的一个热门研究课题。

本文对于如何更高效地使用空间卷积也进行了一些探究。在 ConvNeXt 工作[9]中，作者更加关注调整训练的策略、如何构建模块中的空间卷积以及空间卷积的位置。而本文比较了 ViTs 模型与卷积神经网络模型在编码空间信息的不同方式，并且探索了比 ConvNeXt 工作中更大的卷积核为性能带来的影响。如图 3.1 左侧所示，自注意力机制通过所有其他位置的加权求和来计算特征中每个

像素的输出值。这个过程同样也可以通过计算大卷积核的输出权值和特征值之间的 Hadamard 积来模拟，本文将其称为卷积注意力机制，如3.1右侧所示。若简单直接地使用卷积，在对模型进行训练后其参数为静态不变的，因此对于不同域的输入图像直接使用卷积很难很好的适应；而自注意力机制生成的注意力矩阵由输入内容决定，其可以自由地适应输入内容，即自注意力机制具有数据驱动的特性。本文的卷积注意力机制同样想赋予卷积神经网络动态适应输入图像变换的能力，通过 Hadamard 积能够一定程度上解决这样的困难。本章在多个下游数据集上的实验证明，使用卷积生成权重矩阵经过细致的模型设计之后也可以得到很好的结果。

只需将 ViTs 中的自注意力替换为所提出的卷积注意力操作就能得到本章提出的网络，称之为 Conv2Former。其背后的含义是，本章旨在使用上述的卷积注意力机制构建一种 Transformer 形式的卷积神经网络，在其中卷积得到的特征被用作权重来对值特征进行重新加权。与经典的使用自注意力机制的 ViTs 不同，本章的方法与许多经典的卷积神经网络一样是完全基于卷积的，因此其计算增长是线性的，并不是随着图像分辨率的增加呈现出二次增长的变化。这使得本章提出的方法更适合下游任务，如目标检测和高分辨率语义分割。更有趣的是，本章的方法对于更大的卷积核，如 11×11 和 21×21 的卷积，能够获得更多的收益。这与先前的一些基于卷积神经网络[9,104]的工作中得出的结论不同，它们表明，当使用标准的深度可分离卷积核大于 9×9 的大小时，对下游任务的性能几乎不会带来性能提升，并且同时会增加计算负担。本章还展示了 Conv2Former 优于最近一些使用超大卷积核[29,32]的工作。本章在流行的视觉任务上评估了 Conv2Former 的性能，包括 ImageNet 分类任务[99]，COCO 目标检测/实例分割[94]和 ADE20K 语义分割[92]。为了验证 Conv2Former 在更大的数据集上的性能，本章还在 ImageNet-22k 数据集上进行了预训练，并在下游任务上评估了性能。实验结果表明，Conv2Former 在性能上优于 ConvNeXt[9] 和 EfficientNetV2[20] 等流行的卷积神经网络模型。本章的主要贡献可概括为如下：

- 本章设计了卷积注意力机制，并以卷积注意力机制为基础构建骨干网络 Conv2Former，在多个著名的基准数据集上验证了本文所提出的骨干网络，包括 ImageNet-1k、Microsoft COCO 和 ADE20K，具体来说，在 ImageNet-1k 数据集上本文使用 ImageNet-22k 进行预训练，模型在大约 199M 的参数量情况下能够取得 87.7% 的 Top-1 准确率。

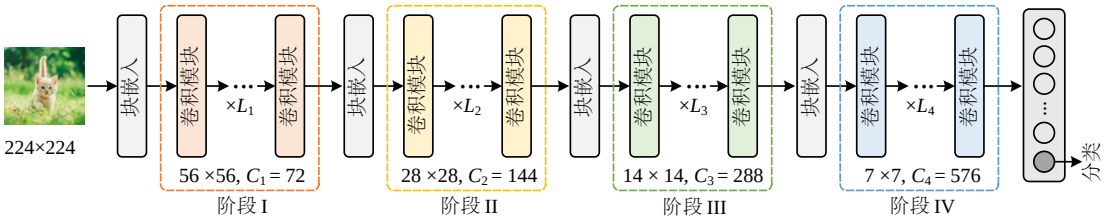


图 3.2 Conv2Former 的总体框架。与前有的卷积神经网络和 Swin-Transformer 一样，本文采用了一个金字塔形式的架构，共包括 4 个阶段，每个阶段包含不同数量的卷积模块。该图展示的模型结构为 Conv2Former-T，其中 $\{L_1, L_2, L_3, L_4\} = \{3, 3, 12, 3\}$。

- 本章探索了不同大小卷积核在 Conv2Former 框架下的性能并给出了分析结论，表明使用卷积注意力机制能够使得网络性能随着卷积核大小的增加而进一步提高，打破了前有工作[9,104]的结论。

第二节 基于卷积自注意力机制的骨干网络设计

本节将详细阐述了本章提出的 Conv2Former 的体系结构，并提供了一些有用的模型设计和网络层数调整的建议。

3.2.1 模型架构设计

总体架构。总体结构已展示在图3.2中。与 ConvNeXt[9]和 Swin-Transformer 网络[6]类似，本文的 Conv2Former 也采用了金字塔结构。其总共包含四个阶段，特征图在不同的阶段具有不同的特征图分辨率。在两个连续的阶段之间，使用补丁嵌入块来降低分辨率，通常使用 2×2 带有步长 2 的卷积来实现。在网络中的不同阶段具有不同数量的卷积模块。本章构建了五个 Conv2Former 变体，即 Conv2Former-N、Conv2Former-T、Conv2Former-S、Conv2Former-B 和 Conv2Former-L。详细参数配置请见表3.1。

表 3.1 本章提出的 Conv2Former 各种变体的简要配置，本文共实现了其 5 种变体，分别含有 15M, 27M, 50M, 90M, 和 199M 的参数量。

模型	$\{C_1, C_2, C_3, C_4\}$	$\{L_1, L_2, L_3, L_4\}$
★ Conv2Former-N	$\{64, 128, 256, 512\}$	$\{2, 2, 8, 2\}$
★ Conv2Former-T	$\{72, 144, 288, 576\}$	$\{3, 3, 12, 3\}$
★ Conv2Former-S	$\{72, 144, 288, 576\}$	$\{4, 4, 32, 4\}$
★ Conv2Former-B	$\{96, 192, 384, 768\}$	$\{4, 4, 34, 4\}$
★ Conv2Former-L	$\{128, 256, 512, 1024\}$	$\{4, 4, 48, 4\}$

表 3.2 三个当前比较流行的模型不同阶段的模块数量配置与最终性能。从表的最后一行不难发现，略微改变卷积模块的数量就能很轻松地提高模型性能。

模型	参数量	FLOPs	阶段配置	Top-1 Acc.
ResNet-50[3]	26M	4.0G	3-4-6-3	78.5%
Swin-T[6]	28M	4.5G	2-2-6-2	81.5%
ConvNeXt-T[9]	29M	4.5G	3-3-9-3	82.1%
★ Conv2Former-N	15M	2.2G	2-2-8-2	81.5%
★ Conv2Former-T	27M	4.4G	3-3-12-3	**83.2%**

每个阶段的详细配置。当可学习参数数量固定时，如何安排网络的宽度和深度对模型最终的性能至关重要[19,26]。原始的 ResNet-50 将每个阶段的模块数量设置为 (3,4,6,3)。为了进一步提高模型的性能，ConvNeXt-T 遵循 Swin-T 使用的原则，将每个阶段的模块数改为 (3,3,9,3)，当构建更大的模型时，每个阶段的模块比例设置为 $1:1:3:1$。不同的是，本文稍微调整了比率，如表 3.1所示。本章观察到，对于一个小型模型（小于 30M 个参数），当网络更深时能够取得更优的性能。表 3.2中可以找到四个不同的小型模型之间的简要比较。

3.2.2 卷积注意力机制模块

本章在每个阶段中使用的卷积模块与 Transformer 中的自注意力模块有着类似的结构。自注意力模块主要包括用于空间编码的自注意力层和用于通道混合的多层感知机。其是一种当下十分流行的空间编码操作，使用矩阵乘法对全局信息进行交互，并且已被证明比局部的卷积更有效。而不同的是，本章用简单的卷积注意力替换了自注意力层，即达到了对全局信息进行建模的效果，又降低了模型的复杂度。

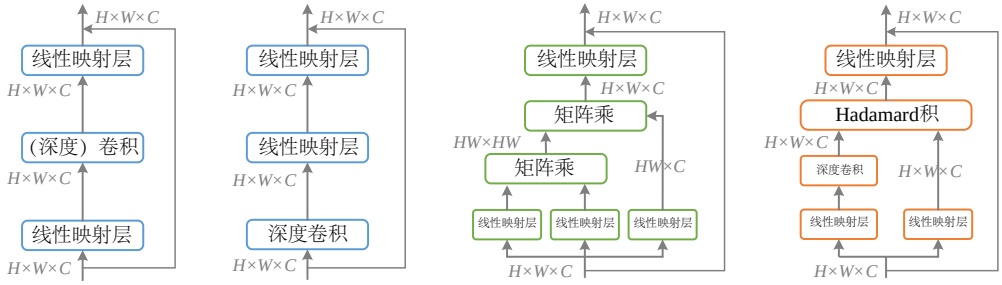
自注意力机制。对于一个长度为 N 的输入 token 序列 $\mathbf{X}$，自注意力机制首先使用线性映射层生成键特征 $\mathbf{K}$，查询特征 $\mathbf{Q}$ 和值特征 $\mathbf{V}$，其中 $\mathbf{X},\mathbf{K},\mathbf{Q},\mathbf{V}\in\mathbb{R}^{N\times C}$，$N=H\times W$，$C$ 是通道数，H 和 W 是输入特征的高度和宽度。基于键特征和查询特征的相似度得分矩阵 $\mathbf{A}$，可用使用其与值特征进行加权平均得到输出，即

$$\text{Attention}(\mathbf{X})=\mathbf{AV}, \tag{3.1}$$

其中 $\mathbf{A}$ 度量的是每对输入 token 之间的关系，可以写成

$$\mathbf{A}=\text{Softmax}(\mathbf{QK}^{\top}). \tag{3.2}$$

需要注意的是，为了简化描述，这里省略了缩放因子。尽管自注意力在编码空



(a) 残差模块[3] (b) ConvNeXt[9] (c) Transformers[5,105] (d) 卷积注意力

图 3.3 基于自注意力与典型的卷积模块中不同的空间编码方式。本章的方法使用深度卷积的输出作为权重来对值特征进行重新加权，如图（d）所示。

间信息方面具有很不错的效果，但相似度得分矩阵 $\mathbf{A}$ 的空间形状为 $\mathbb{R}^{N\times N}$，使得自注意力的计算复杂度随着序列长度 N 的增加呈平方级增长。

卷积注意力机制。在自注意力模型中，相似性矩阵 A 可以通过式3.2计算得到。然而在本文的卷积注意力中，采用卷积特征来对值特征 $\mathbf{V}$ 进行重新加权，而不是通过计算相似性矩阵 $\mathbf{A}$，以此来简化自注意力模型。

具体来说，给定输入符号 $\mathbf{X}\in\mathbb{R}^{H\times W\times C}$，本章使用一个简单的大小为 $k\times k$ 的深度卷积和 Hadamard 乘积来计算输出 $\mathbf{Z}$，具体如下：

$$\mathbf{Z}=\mathbf{A}\odot\mathbf{V}, \tag{3.3}$$

$$\mathbf{A}=\mathrm{DConv}_{k\times k}(\mathbf{W}_1\mathbf{X}), \tag{3.4}$$

$$\mathbf{V}=\mathbf{W}_2\mathbf{X}, \tag{3.5}$$

其中 $\odot$ 是 Hadamard 乘积运算符，$\mathbf{W}_1$ 和 $\mathbf{W2}$ 是两个线性映射层的权重矩阵，DConv$k\times k$ 表示大小为 $k\times k$ 的深度卷积。上述卷积注意力操作使得每个空间位置 (h,w) 能够与中心点为 (h,w) 的 $k\times k$ 正方形区域内的所有像素相关联。与此同时，通道间的信息交互可以通过线性映射层实现。每个空间位置的输出是正方形区域内所有像素的加权和。

优势。图 3.3 展示了残差模块、自注意力机制和本章提出的卷积注意力之间的图示比较。与自注意力相比，本章的方法利用卷积来构建全局的上下文关系，特别是在处理高分辨率图像时，这比自注意力机制更具有内存效率。同时，不同于自注意力机制需要大量数据进行训练，卷积因其局部特性在数据量较小的情况下也能快速收敛。与经典的残差模块[3,9]相比，本章的方法通过注意力操作同样能够对网络的输入进行自适应的改变。

3.2.3 微观设计

使用比 7×7 更大的卷积。自从 VGGNet[11]和 ResNets[3,15]出现以来，3×3 卷积已成为构建卷积神经网络的标准选择，残差模块被广泛应用。早期的研究工作[1,12]尝试使用更大的卷积核，但大卷积核的计算开销是一个严重的问题。后来，深度可分离卷积的出现[106]改变了这种情况。ConvNeXt 表明，将卷积神经网络的卷积核大小从 3 扩大到 7 可以提高分类性能，但是继续提升卷积核大小时性能趋向于饱和状态。然而，进一步增加卷积核大小几乎没有性能增益，而且如果没有使用重参数化技术会带来计算负担[29,107]。

本文认为，让 ConvNeXt 从更大的卷积核中并没有收益的原因是使用空间卷积的方式。对于 Conv2Former，本章观察到当卷积核大小从 5×5 增加到 21×21 时，性能几乎始终有所提升。这种现象不仅发生在 Conv2Former-T（$82.8 \rightarrow 83.4$）上，而且在 80M+ 参数的 Conv2Former-B 上也成立（$84.1 \rightarrow 84.5$）。考虑到模型效率，本文默认将卷积核大小设置为 11×11。

权重策略。如图 3.3（d）所示，本文将深度卷积的输出视为线性映射之后的特征的权重，用于对特征进行重新加权后输出。值得注意的是，在 Hadamard 乘积之前，本文没有使用激活函数或归一化层（例如 Sigmoid 函数或 L_p 归一化）。这在本文的工作中是获得良好性能的重要因素。前有工作通常会使用 Softmax 函数或 Sigmoid 函数将生成的权重规整化到 0 至 1 的区间范围，例如 ViT[5]中使用 Softmax 函数对相似度矩阵进行归一化。而对于本文的注意力机制，使用类似的归一化将会带来部分的性能退化。具体来说，若在本文中添加像 SENet[16]中所使用 Sigmoid 函数会使性能下降超过 0.5%。

需要强调的是，FocalNet[108]采用了与本文类似的加权策略，但其动机不同。FocalNet 旨在通过 3×3 深度可分离卷积和全局平均池化来提取多级特征，以进行多层级的上下文聚合。与此不同的是，本章尝试通过利用简单的大核卷积来简化自注意力操作，并探索一种有效利用大核空间卷积用于卷积神经网络的方法。本章的方法相较于 FocalNet 更加简单，并且在各个基准数据集上的实验都证明了本章 Conv2Former 相对于 FocalNet 的优势。

标准化和激活函数。本文在归一化层上基本遵循了原始的 ViT 和 ConvNeXt，采用了层归一化[109]而不是广泛使用的批归一化[110]。在激活层方面，本文使用 GELU[111]激活函数。本文发现层归一化和 GELU 的组合可以带来 0.1%-0.2% 的性能提升。

表 3.3 Conv2Former 的不同变体在 ImageNet-1k/22k 数据集上训练使用的随机深度率。表中 “dpr.” 表示随机深度率。

模型	数据集	dpr.
Conv2Former-N/T/S/B/L	ImageNet-1k	0.1/0.1/0.2/0.7/0.7
Conv2Former-S/B/L	ImageNet-22k	0/0.1/0.1

表 3.4 Conv2Former 的不同变体在 ImageNet-1k 数据集上微调使用的随机深度率。表中 “dpr.” 表示随机深度率。

模型	分辨率	dpr.
Conv2Former-S/B/L	224×224	0.1/0.2/0.3
Conv2Former-B/L	384×384	0.2/0.3

表 3.5 Conv2Former 的不同变体在 COCO 数据集上微调使用的随机深度率。

检测模型	Mask R-CNN	Cascade Mask R-CNN
骨干网络	Conv2Former-T	Conv2Former-T/S/B
dpr.	0.2	0.2/0.5/0.8

表 3.6 Conv2Former 的不同变体在 ADE20K 数据集上微调使用的随机深度率。

模型	预训练数据集	dpr.
Conv2Former-T/S/B	ImageNet-1k	0.2/0.3/0.5
Conv2Former-L	ImageNet-22k	0.4

第三节 实验对比以及结果分析

3.3.1 实验设置

3.3.1.1 数据集

本文在广泛使用的 ImageNet-1k 数据集[99]上评估了所提出的 Conv2Former 的分类性能，该数据集包含约 1.2M 张训练图像，共 1,000 个不同的类别。本文报告了总共 50k 张图片的验证集上的结果。为了验证本文的 Conv2Former 在数据可扩展性方面的能力，本章仿照其他一些流行的模型[6,9]，使用大规模的 ImageNet-22k 数据集对所提出的 Conv2Former 进行了的预训练以及下游任务微调测试。

ImageNet-22k 数据集包含约 14M 张图像，共 21,841 个类别。不同于 ImageNet-1k 数据集，ImageNet-22k 数据集中的图片含有一定的噪声，并且存在类别分布不均衡的问题，训练更加具有挑战性。在预训练之后，本章使用 ImageNet-1k 数据集进行微调，并在 ImageNet-1k 验证集上报告结果。除此以外，为了测试本文提出的骨干网络的泛化能力，本章挑选了另外的两个下游任务：目标检测/实例分割和语义分割，使用 COCO 数据集[94]和 ADE20K 数据集[92]对模型进行微调并测试最终的性能。

3.3.1.2 训练设置

本文的实现基于 `PyTorch` 库[114]以及 `timm` 库[115]。在 ImageNet-1k 训练过程中，本文设置批大小为 1024，使用 AdamW 优化器[116]，对于学习率，本章采用线性学习率缩放策略 $lr = \text{LRbase} \times \text{batch\_size}/1024$，参照先前工作的建

表 3.7 在 ImageNet 数据集[99]上 Top-1 准确率比较结果。与先前流行的 Transformer 和卷积神经网络相比，本文的 Conv2Former 在不同模型大小的网络变体上都取得了出人意料的良好结果。

模型	参数量	FLOPs	图像大小	Top-1 准确率
ResNet18[3]	12M	1.8G	224×224	69.8%
PVT-Tiny[40]	13M	1.9G	224×224	75.1%
PoolFormer-S12[8]	12M	2.0G	224×224	77.2%
VAN[31]	14M	2.5G	224×224	81.1%
★ Conv2Former-N	15M	2.2G	224×224	**81.5%**
ResNet50-d[3,78]	26M	4.3G	224×224	79.5%
SwinT-T[6]	28M	4.5G	224×224	81.5%
ConvNeXt-T[9]	29M	4.5G	224×224	82.1%
Focal-T[41]	29M	4.9G	224×224	82.2%
★ Conv2Former-T	27M	4.4G	224×224	**83.2%**
SwinT-S[6]	50M	8.7G	224×224	83.0%
ConvNeXt-S[9]	50M	8.7G	224×224	83.1%
NFNet-F0[26]	72M	12.4G	256×256	83.6%
★ Conv2Former-S	50M	8.7G	224×224	**84.1%**
DeiT-B[38]	86M	17.5G	224×224	81.8%
RegNetY-16G[27]	84M	16.0G	224×224	82.9%
RepLKNet-31B[29]	79M	15.3G	224×224	83.5%
SwinT-B[6]	88M	15.4G	224×224	83.5%
ConvNeXt-B[9]	89M	15.4G	224×224	83.8%
FocalNet-B[108]	89M	15.4G	224×224	83.9%
MOAT-2[112]	73M	17.2G	224×224	84.2%
EffNet-B7[113]	66M	37.0G	600×600	84.3%
★ Conv2Former-B	90M	15.9G	224×224	**84.4%**

议，初始学习率 LRbase 设置为 0.001，权重衰减率设置为 5×10^{-2}，预热步长设置为 5。在 ImageNet-1k 实验中，本章将图像大小随机裁剪为 224×224，并采用一些常见的数据增强方法，如 MixUp[117]和 CutMix[118]。同时还使用随机深度[119]，随机擦除[120]，标签平滑[13]，随机增强[121]和初始值为 1e-6 的层放缩策略[55]。本文将所有模型训练 300 个步长。在 ImageNet-22k 上的实验中，同样遵循 ConvNeXt[9]的训练方式，本章首先在此数据集上对模型进行 90 个步长的预训练，然后再在 ImageNet-1k 数据集上进行 30 个步长的微调。在 ImageNet-22k 数据集上进行预训练过程时，由于数据集本身带有一定的噪声，为了稳定训练，设置预热步长为 5，整体的批大小设置为 4096，学习率设置为 0.004。当在

表 3.8 在 ImageNet-22k 数据集上预训练的情况下，本章的模型在 ImageNet 验证集上得到了如下的 Top-1 准确率结果。与 ConvNeXt 相比，Conv2Former 表现出了稳定的提升。此外，本章的 Conv2Former-L 模型的性能也优于 EfficientNetV2-XL 和 CoAtNet-3。

模型	参数量	FLOPs	图像大小	Top-1 准确率
ConvNeXt-S[9]	50M	8.7G	224×224	84.6%
★ Conv2Former-S	50M	8.7G	224×224	**84.9%**
SwinT-B[6]	88M	15.4G	224×224	85.2%
ConvNeXt-B[9]	89M	15.4G	224×224	85.8%
MOAT-2[112]	73M	17.2G	224×224	86.0%
★ Conv2Former-B	90M	15.9G	224×224	**86.2%**
ViT-B/16[5]	87M	55.5G	384×384	85.4%
SwinT-B[6]	88M	47.0G	384×384	86.4%
EffNet-V2-L[20]	120M	53.0G	480×480	86.8%
ConvNeXt-B[9]	89M	45.1G	384×384	86.8%
★ Conv2Former-B	90M	46.7G	384×384	**87.0%**
EffNet-V2-XL[20]	208M	94.0G	480×480	87.3%
SwinT-L[6]	197M	34.5G	224×224	86.3%
ConvNeXt-L[9]	198M	34.4G	224×224	86.6%
MOAT-3[112]	190M	44.9G	224×224	86.8%
★ Conv2Former-L	199M	36.0G	224×224	**87.0%**
SwinT-L[6]	197M	104G	384×384	87.3%
ConvNeXt-L[9]	198M	101G	384×384	87.5%
CoAtNet-3[49]	168M	107G	384×384	87.6%
★ Conv2Former-L	199M	105.9G	384×384	**87.7%**

ImageNet-1k 数据集上进行微调时，权重衰减率设置为 1×10^{-8}，学习率设置为 5×10^{-5}，批大小设置为 512，并且不使用 MixUp[117]和 CutMix[118]数据增强方法。

与 ConvNeXt[9]相比，本章没有使用逐层的学习率衰减策略[59]以及参数指数滑动平均策略，因为本章发现它们对 Conv2Former 的训练没有帮助，并且逐层学习率衰减策略需要对一些超参数进行人工的精细调整。表 3.3展示了 Conv2Former 的不同变体在（预）训练中使用的随机深度率。表 3.4展示了 Conv2Former 的不同变体在微调过程中使用的随机深度率。

当在下游任务数据集 COCO[94]与 ADE20K[92]上进行训练时，本章基本遵循[9,59]与[6]的训练参数与策略。在表 3.5与表 3.6中展示了本章的 Conv2Former 的不同模型变体在这些数据集上的随机深度率。

表 3.9 Conv2Former 与使用不同卷积核大小的最新卷积神经网络的性能比较。从表中可以看到，在没有任何其他训练技术（如重新参数化或使用稀疏权重）的情况下，本文的的 Conv2Former 使用 11×11 卷积核大小取得了最佳结果。这些实验表明，本章的卷积自注意力操作可以更有效地编码空间信息。更多不同卷积核大小的结果可以在节3.3.3中找到。

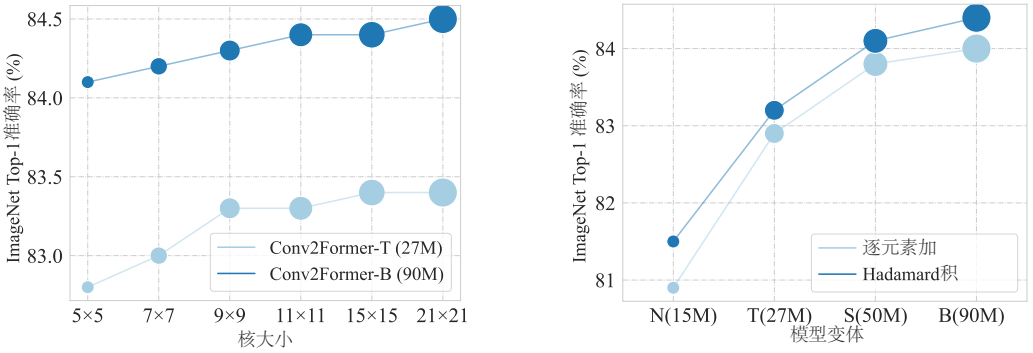
模型	核大小	参数量	FLOPs	Acc.
RepLKNet-31B[29]	31×31	79M	15.3G	83.5%
ConvNeXt-B[9]	7×7	89M	15.4G	83.8%
SLaK-B[32]	51×51	95M	17.1G	84.0%
★ Conv2Former-B	7×7	89M	15.6G	84.2%
★ Conv2Former-B	11×11	90M	15.9G	**84.4%**

3.3.2 与其他方法的比较结果

本文将 Conv2Former 与一些当先流行的骨干网络架构进行了性能比较，包括 Swin-Transformer[6]、ConvNeXt[9]、PoolFormer[8]、ViT[5]、VAN[31]、NFNet[26]、DeiT[38]、RegNet[27]、FocalNet[108]、EfficientNets[19-20]、CoAtNet[49]、RepLKNet[29]和 MOAT[112]。值得注意的是，其中一些是卷积神经网络和 Transformer 的混合模型，一般情况下，将卷积融入到 Transformer 网络中使得网络即能过处理局部信息，同时能够融合全局信息，对网络性能十分利好。本文使用的是纯卷积构建的网络，并未加入注意力机制，旨在使用卷积构建性能最先进的骨干网络。

ImageNet-1k 数据集。本章首先在 ImageNet-1k 数据集上训练了 Conv2Former，并在表 3.7中展示了在验证集上的结果。对于小型模型（< 30M），相较于 ConvNeXt-T 和 SwinT-T，本文的 Conv2Former 分别具有 1.1%和 1.7%的性能增益。即使是参数只有 15M，FLOPs 为 2.2G 的 Conv2Former-N，其也能够与参数为 28M，FLOPs 为 4.5G 的 SwinT-T 性能相当。对于参数量稍大的 Conv2Former-B，性能上的增益稍稍减小，但与 ConvNeXt-B 和 SwinT-B 相比仍然有 0.6%和 0.9%的提高。与其他的流行模型相比，本章的 Conv2Former 在具有相似模型大小的情况下，性能表现更好。值得注意的是，本章的 Conv2Former-T 甚至表现比 EfficientNet-B7（84.4%v.s. 84.3%）更好，而后者的计算量是 Conv2Former-T 的两倍（37G v.s. 15G）。

ImageNet-22k 数据集。为了反映本文的 Conv2Former 具有数据扩展能力，本文在大规模的 ImageNet-22k 数据集上对 Conv2Former 进行了预训练，随后在 ImageNet-1k 数据集上进行微调。在所有实验中，本章遵循[9]中使用的设置来训



（a）不同核大小的实验结果；　　（b）不同融合策略的实验结果。

图 3.4　消融实验。对于 Conv2Former-T 和 Conv2Former-B，当将卷积核的大小从 5×5 增加到 21×21 时，能够很显然观察到一致的性能提升。对于融合策略，当将 Hadamard 积替换为逐元素求和操作时，Conv2Former 的四种模型变体的性能都会下降。

练和微调模型。结果已展示在表 3.8中。与 ConvNeXt 的不同变体相比，本章的 Conv2Former 在模型大小相似的情况下都表现更好。通常，本章的 Conv2Former-T 比 ConvNeXt-B 和 MOAT-2 网络表现更好，并且后者比 Conv2Former 的需要的计算量更多。此外，从表中可以发现，当在更大的分辨率 384×384 上微调时，本章的 Conv2Former-L 相较于混合模型 CoAtNet 和 MOAT 取得了更好的性能。本章的 Conv2Former-L 取得了最佳结果 87.7%。

讨论。采用大卷积核是协助卷积神经网络构建长距离依赖关系的一种简单直接的方法。然而，直接在现有的一些基于 CNN 的架构中使用大卷积核（$> 7 \times 7$）会使识别模型难以被优化[9,104]。近期有一些研究致力于开发新技术来使得大卷积核与 CNN 更加适配。表 3.9展示了最近的一些使用不同核大小的大卷积核工作以及它们与 Conv2Former 的比较结果。从表 3.9可以发现，在不添加任何其他训练技巧（例如重新参数化或使用稀疏权重）的情况下，本章的 Conv2Former 在基础的训练配置下使用 7×7 的核大小已经优于表中的其他方法。当使用更大的核 11×11，Conv2Former 能够取得更好的性能提升。这些结果反映了本章的卷积注意力机制的优势。

3.3.3　方法分析

核大小。在 ConvNeXt 的工作中表明，当深度卷积的核大小超过 7×7 时，模型的性能不会有所提升。本节研究了当使用更大的卷积核时，Conv2Former 模型性能会如何变化。本节选择了 6 个不同的大小的卷积核，即 5×5、7×7、9×9、11×11、15×15 和 21×21，并展示了基于两个模型变体 Conv2Former-T

表 3.10 卷积注意力中采用不同融合策略的性能表现。表中所有结果都基于 Conv2Former-T 模型。从表中可以发现，使用简单的 Hadamard 积获得了最好的结果。

权重策略	Top-1 准确率
逐元素加	82.7%
在 **A** 后添加 Sigmoid 函数	82.3%
在 **A** 后添加 L_1 归一化	82.8%
将 **A** 的值线性归一化到 $(0,1]$	82.2%
★ Hadamard 积	**83.2%**

和 Conv2Former-T 的结果，如图3.4（a）所示。当卷积核的大小增加到 21×21 时，性能的提升似乎达到了饱和的状态。然而这个发现与 ConvNeXt 得出的结论有很大不同，后者认为使用大于 7×7 的卷积核不会带来明显的性能提升。这表明，使用式3.3中表达的卷积特征作为卷积注意力的权重，比传统的一些直接利用局部卷积的方法[3,9]，能够更有效地利用大型卷积核。

Hadamard 积优于加法。如图 3.3（d）所示，本章使用深度卷积提取的卷积特征作为右侧线性分支的权重，通过 Hadamard 积融合操作对右侧的特征进行重新加权。在本章的实验中，同时还尝试使用逐元素求和作为策略来两个分支进行融合操作。图 3.4（b）显示了本章的 Conv2Former 在不同模型尺寸下的比较结果。结果表明，Hadamard 积的效果优于逐元素求和，表明在 Conv2Former 这样的全卷积网络情况下，卷积注意力机制在编码空间信息方面比逐元素求和更加有效。与此同时，从图 3.4（b）能够观察到 Hadamard 积对于小型模型能够带来更多的益处。

权重策略。除了上文提到的两种融合策略，本章还尝试了其他融合策略来对特征图进行操作，包括在 **A** 后加入 Sigmoid 函数，对 **A** 进行 L_1 规范化以及线性规范化 **A** 的值到 $(0,1]$。实验结果总结在表 3.10中。从表中可以看到，Hadamard 积相较于所有其他操作能够产生更好的结果。更有趣的是，当使用 Sigmoid 函数或将 **A** 的值线性归一化到 $(0,1]$ 时，性能会下降更多。这与传统的注意机制（例如 SE[16]和 CA[18]）不同，对于传统的注意力机制，其一般使用 Sigmoid 函数将注意力特征归一化至 $(0,1]$ 区间内，使得其起到放缩的效果，而本文的卷积注意力并不需要这样的归一化操作。本章将其留待未来研究。

3.3.4 各向同性模型的结果

传统的 CNN 模型[1,3,11]采用多层级结构，而由于自注意力机制的计算量比较大，经典的 ViT 模型[5,38]采用十分简单的结构，包含一个块嵌入层和一

表 3.11 各向同性的 Conv2Former、ConvNeXt 与 ViT 的比较结果。“3 Convs” 代表着在网络开头使用了三个卷积层进行块嵌入操作，正如[49,53,58] 中所做的那样。本章的 Conv2Former 在具有类似的参数和计算量的情况下取得了更好的结果。

模型	块嵌入方式	参数量	FLOPs	Top-1 准确率
DeiT-S	1 Conv	22M	4.6G	79.8%
ConvNeXt-IS	1 Conv	22M	4.3G	79.7%
★ Conv2Former-IS	1 Conv	23M	4.3G	81.2%
★ Conv2Former-IS	3 Convs	23M	4.5G	82.0%
DeiT-B	1 Conv	87M	17.6G	81.8%
ConvNeXt-IB	1 Conv	87M	16.9G	82.0%
★ Conv2Former-IB	1 Conv	86M	16.5G	82.7%
★ Conv2Former-IB	3 Convs	87M	17.3G	83.0%

系列具有相同 token 序列长度的 Transformer 模块。这种简单的结构在最近的 Transformer 相关工作中得到了非常广泛的应用。本文遵循 ConvNeXt[9]的设置，对 Conv2Former 在 ViT 式的风格架构下的表现进行了研究。类似于 ConvNeXt，本章将 Conv2Former-IS 和 Conv2Former-IB 中的模块数都设置为 18，并调整通道数以匹配模型大小。本章研究了两种版本的块嵌入模块：一个 16×16 的卷积层，步长为 16；三个卷积层，同[53]一致。表 3.11展示了实验结果。这里以 DeiT-S 和 DeiT-B 模型为基线模型。为了简洁起见，在模型名称中添加字母 “I”，表示相应的模型使用与原始 ViT 相同的各向同性架构。从表中可以看到，对于参数约为 22M 的小型模型，本章的 Conv2Former-IS 表现比 DeiT-S 和 ConvNeXt-IS 要好得多，性能提高约为 1.5%。当将模型大小扩展到 80M+ 时，本章的 Conv2Former-IB 实现了 82.7% 的 Top-1 准确率，比 ConvNeXt-IB 高 0.7%，比 DeiT-B 高 0.9%。此外，使用三个卷积作为块嵌入模块进行下采样可以进一步提高结果。

3.3.5 下游任务结果

COCO 数据集上的结果。 MSCOCO[94]是一个用于目标检测的大型数据集，其一共包含 80 个类别。本节采用了两种流行的目标检测器，Mask R-CNN[122]和 Cascade Mask R-CNN[123]，并报告了物体检测和实例分割的结果。当训练模型时，本节遵循 ConvNeXt 工作[9]中使用的实验设置，包括使用多尺度的训练方式、AdamW 优化器、3× 的学习策略、GIoU 损失[124]等。读者可以参考[9,59]获取更详细的实验设置信息。本章使用 MMDetection 库[125]运行所有目标检测实验。结果如表 3.12所示。对于 Tiny 模型变体，当使用 Mask R-CNN 框架进行目标检测

表 3.12 使用 Mask R-CNN[122]与 Cascade Mask R-CNN[123]在 COCO[94]数据集上进行目标检测与实例分割的结果。注意，表中的模型在 ImageNet-1k 数据集上进行了预训练。

模型	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mask}	AP^{mask}_{50}	AP^{mask}_{75}
Mask R-CNN[122] 3× schedule						
SwinT-T	46.0	68.1	50.3	41.6	65.1	44.9
ConvNeXt-T	46.2	67.9	50.8	41.7	65.0	44.9
★ Conv2Former-T	**48.0**	69.5	52.7	**43.0**	66.8	46.1
Cascade Mask R-CNN[123] 3× schedule						
SwinT-T	50.4	69.2	54.7	43.7	66.6	47.3
ConvNeXt-T	50.4	69.1	54.8	43.7	66.5	47.3
SLaK-T	51.3	70.0	55.7	44.3	67.2	48.1
★ Conv2Former-T	**51.4**	69.8	55.9	**44.5**	67.4	48.3
SwinT-S	51.9	70.7	56.3	45.0	68.2	48.8
ConvNeXt-S	51.9	70.8	56.5	45.0	68.4	49.1
★ Conv2Former-S	**52.8**	71.4	57.3	**45.7**	69.0	49.8
SwinT-B	51.9	70.5	56.4	45.0	68.1	48.9
ConvNeXt-B	52.7	71.3	57.5	**45.6**	69.0	49.8
★ Conv2Former-B	**52.8**	71.1	57.2	**45.6**	68.7	49.3

时，本章的 Conv2Former-T 相比 SwinT-T 和 ConvNeXt-T 获得了约 2% 的 AP 提升。对于实例分割，性能提升也超过了 1%。当使用 Cascade Mask R-CNN 框架时，本章观察到比 SwinT-T 和 ConvNeXt-T 多超过 1% 的性能提升。当扩大模型的规模，使用更大计算量的模型时，Conv2Former 的性能提升也是比较明显的。

ADE20K 数据集上的结果。 ADE20K[92]是一个被广泛使用的语义分割数据集，它包含 150 个类别和各种场景，具有 1038 个图像级标签，相较于其他常用的语义分割基准数据集难度更大。按照[6,9]的做法，本章使用训练集对模型进行训练，并在验证集上报告结果。对于 Conv2Former-T、Conv2Former-S、Conv2Former-B 三个变体模型，本章将输入图像随机裁剪为 512 × 512，对于 Conv2Former-L 模型，本章使用 640 × 640 的裁剪大小来进一步提高模型的性能。本章使用 UperNet[126]作为分割头。结果总结如表 3.13。对于不同规模的模型，本章的 Conv2Former 可以优于 Swin Transformer 和 ConvNeXt。值得注意的是，在 Tiny 这样的模型规模下，相对于 ConvNeXt，有 1.3% 的 mIoU 提升，在 Base 的模型规模下，提升为 1.1%。当进一步增加模型大小时，本章的 Conv2Former-L 与 UperNet 结合实现了 54.3% 的 mIoU 得分，也明显优于 Swin-L 和 ConvNeXt-L。

表 3.13 与 Swin-T 和 ConvNeXt 在 ADE20k 数据集[92]上的比较结果。本章在所有结果中使用 UperNet[126]作为分割头。在所有模型大小上，本章的 Conv2Former 均取得了最佳结果。

模型	预训练模型	裁剪大小	参数量	mIoU (%)
SwinT-T	ImgNet-1k	512^2	60M	45.8
ConvNeXt-T	ImgNet-1k	512^2	60M	46.7
★ Conv2Former-T	ImgNet-1k	512^2	56M	**48.0**
SwinT-S	ImgNet-1k	512^2	81M	49.5
ConvNeXt-S	ImgNet-1k	512^2	82M	49.6
★ Conv2Former-S	ImgNet-1k	512^2	79M	**50.3**
SwinT-B	ImgNet-1k	512^2	121M	49.7
ConvNeXt-B	ImgNet-1k	512^2	122M	49.9
★ Conv2Former-B	ImgNet-1k	512^2	120M	**51.0**
SwinT-L	ImgNet-22k	640^2	234M	53.5
ConvNeXt-L	ImgNet-22k	640^2	235M	53.7
★ Conv2Former-L	ImgNet-22k	640^2	231M	**54.3**

第四节 本章小结

本章提出了一种新的用于图像理解的纯卷积骨干网络架构，称为 Conv2Former。本章所提出的 Conv2Former 其核心是卷积注意力操作，该操作通过仅仅使用卷积和 Hadamard 积对传统的自注意力机制进行了很好的简化。通过使用卷积注意力操作，本章的骨干网络能够非常有效地利用更大的卷积核（$\geq 7 \times 7$），并且随着卷积核的增大，能够为性能带来接近线性的收益，当卷积核大小扩展到 21×21 时，性能接近饱和。Conv2Former 作为骨干网络，在 ImageNet-1k 数据集上的分类、COCO 数据集上的目标检测和 ADE20K 数据集上的语义分割等领域的实验结果也表明 Conv2Former 的表现相较于以前的基于卷积神经网络的模型和大多数基于 Transformer 的模型要好，甚至优于卷积神经网络与 Transformer 的混合模型。

3.4.1 讨论

近期的最先进视觉识别模型[97,128] 在低层级特征编码方面严重依赖于卷积操作。然而对于基于卷积神经网络的图像理解模型，仍有很大的提升空间。本章认为，需要更深入地探讨以下方面：

首先，如何更有效地利用大型卷积核（$\geq 7 \times 7$）进行图像特征的提取和编

表 3.14 Conv2Former 与 ConvNeXt 的不同变体在实际推理时的推理速度。注意，遵循[127]，本章使用 timm[115]在 NVIDIA V100 TENSOR CORE GPU 上进行推理测试。

模型	参数量	FLOPs	推理速度	Top-1 准确率
ConvNeXt-T	29M	4.5G	760 图片/秒	82.1%
Conv2Former-T	27M	4.4G	585 图片/秒	83.2%
ConvNeXt-S	50M	8.7G	455 图片/秒	83.1%
Conv2Former-S	50M	8.7G	336 图片/秒	84.1%
ConvNeXt-B	89M	15.4G	288 图片/秒	83.8%
Conv2Former-B	90M	15.9G	224 图片/秒	84.4%

码。大型卷积核相比较小卷积核具有更大的感受野，能够提取更全面的局部特征，但是也会导致计算量的增加和模型的复杂度判断的增加。因此，如何应对这种矛盾性，是需要进一步研究的问题。

其次，如何在卷积操作中更有效地捕获大的感受野。在图像理解模型中，仍需要对大的感受野进行更加深入的研究，以提高模型对于更广泛范围的图像场景的感知认知。而对于基于卷积神经网络的模型来说，如何使用具有固定卷积核大小的卷积操作来更好地提取这些特征，是一个值得研究的方向。

最后，如何更有效地引入轻量级的注意力机制到卷积神经网络中。注意力机制是近年来被广泛应用于图像理解领域的一种技术，它能够筛选出图像中最关键的信息，从而提高模型对于重要信息的关注度。如何在卷积神经网络中引入轻量级的注意力机制，以提高模型的准确性和效率，也是值得这方面的研究方向。

3.4.2 局限性

本文旨在研究如何更有效地利用大卷积核。因此本文仅仅关注基于卷积神经网络的模型设计而并未考虑目前最流行的基于 Transformer 的架构改进，如何将所提出的卷积注意力与 Transformer 相结合需要未来更多的探索。与此同时，本文测试了在实际推理过程中的运行速度如表 3.14所示，由于目前 Pytorch 框架对深度卷积的优化问题，当在 Conv2Former 每个模块中使用更多的卷积层时，在 GPU 上的运行速度相较于 ConvNeXt 慢了约 20%-30%。

第四章　基于卷积注意力机制的语义分割

近年来，随着 DETR[129]和 ViT[5]等模型的发展，Transformer 模型不仅在自然语言处理场景下成功应用，而且也逐渐成为计算机视觉领域中各个方向的主流模块之一，其自注意力模块由于能够解决长距离依赖问题而备受视觉研究者的关注。在各大基准数据集上，例如 ImageNet-1k[99]，COCO[94]和 ADE20K[92]等，基于 Transformer 的网络架构其性能已然逐渐超越了基于 CNN 的网络架构。不同于先前的基于自注意力模块的工作，本章在第三章工作的卷积注意力基础之上进行略微修改，提出了一种新的多尺度卷积注意力机制，旨在利用卷积操作来实现类似于注意力机制的功能。相较于以前的工作，卷积注意力机制不仅计算复杂度较低，而且能够捕捉多个不同尺度的信息。Peng 等人[130]同样提出了对卷积进行 $k \times 1$ 和 $1 \times k$ 的分解形式，并利用大核卷积来完成语义分割任务，展现了大核卷积的优势。而与之不同的是，本章工作旨在利用多个不同核大小的卷积来完成卷积注意力机制，并且根据语义分割任务的特点，在网络架构方面，进行了精心的设计，为语义分割任务打造了一个强有力的骨干网络。

本章节的内容安排如下：首先明确了本章研究的动机和贡献，然后详细介绍了基于多尺度卷积自注意力机制的语义分割方法。最后，本章在六个公开语义分割数据集上进行了实验，其中包括 PASCAL VOC 2012[131]、Cityscapes[93]、COCO-Stuff[94]、ADE20K[92]等，同时对实验结果进行了分析和讨论。总之，本章节提出了一种基于多尺度卷积自注意力机制的语义分割方法，能够在较小的计算负荷下将图像信息转换为精确的像素级分割结果。本章节工作为语义分割领域的进一步发展提供了一个有价值的思路和方法。

第一节　研究动机以及贡献

作为计算机视觉中最基本的研究课题之一，语义分割在过去十年中引起了极大的关注，该任务旨在为每个像素分配一个语义类别。从早期基于 CNN 的模型，如 FCN[60]和 DeepLab 系列[70,132-133]，到最近基于 Transformer 的方法，如 SETR[71]和 SegFormer[72]，语义分割模型在网络架构方面经历了重大变革。通过重新审视以前成功的语义分割工作，本文总结了不同模型所拥有的几个关键属

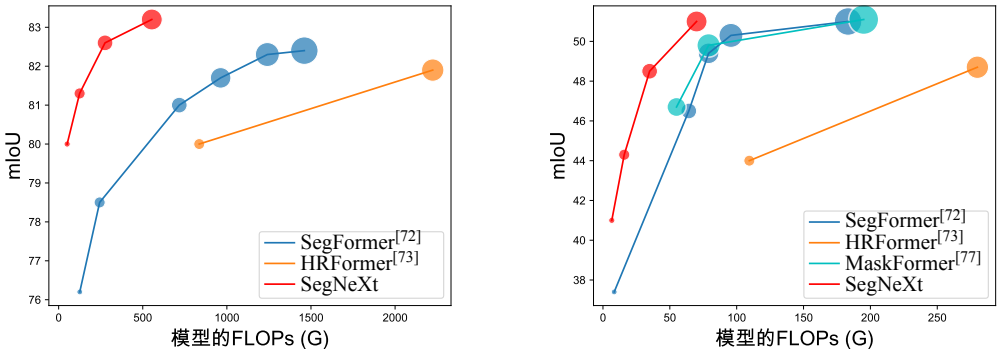


图 4.1 在 Cityscapes（左）和 ADE20K（右）验证集上的性能-计算量曲线。计算 FLOPs 时，Cityscapes 数据的输入尺寸为 2,048 × 1,024，ADE20K 数据为 512 × 512。图中，圆圈的大小表示参数量，更大的圆圈意味着模型拥有更多的参数。从图中不难看出，本章节提出的 SegNeXt 实现了分割的性能和计算的复杂度之间的最佳权衡。

性，如表 4.1所示。基于上述观察，本文认为一个成功的语义分割模型应该具有以下特征：（i）强大的骨干网络作为编码器：与传统的基于卷积神经网络的模型相比，基于 Transformer 的模型的性能提高主要来自于性能更强大的骨干网络；(ii) 多尺度信息交互：与图像分类任务大多识别单一物体不同，语义分割是一个像素级别的密集预测任务，对于同一张图片中不同大小的物体都需要做精细的分割。(iii) 空间注意力：空间注意力允许模型通过对语义区域内的区域进行优先排序来进行分割。(iv) 低计算复杂度：这在处理遥感和城市场景的高分辨率图像时尤其关键。

考虑到上述分析，本章重新思考了卷积注意力的设计，并提出了一个高效而有效的用于语义分割的编码器-解码器架构。前有的基于 Transformer 的模型大多会在解码器中使用卷积来更加细化地提取特征，而本章的方法会将前有的方法倒置，使用更加强大的编码器来提取特征。具体来说，对于编码器中的每个模块，本章对传统卷积进行重新的设计，利用多尺度的卷积特征，通过一个

表 4.1 本章从成功的语义分割方法中观察到有利于提高模型性能的特性。表中，n 指的是像素或 token 的数量。

属性	DeepLabV3+	HRNet	SETR	SegFormer	SegNeXt
强大的骨干网络	✗	✗	✓	✓	✓
多尺度的交互	✓	✓	✗	✗	✓
全局的空间注意力	✗	✗	✓	✓	✓
计算复杂度	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$

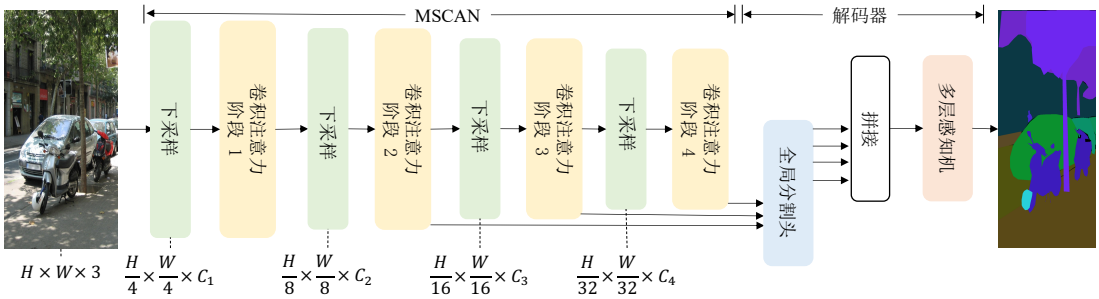
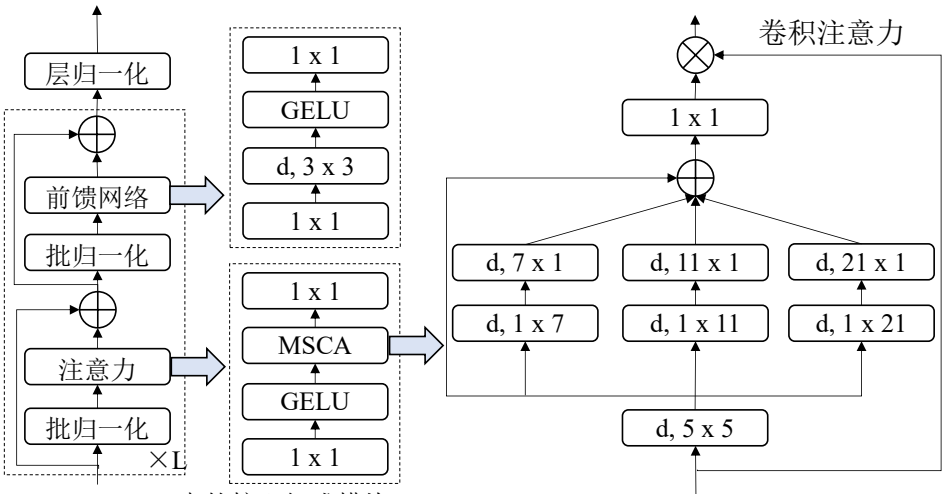


图 4.2 SegNeXt 网络的整体架构，整个网络总体遵循编码器-解码器架构。对于编码器部分，本文使用多层级的、基于多尺度卷积注意力机制的 MSCAN 网络；对于解码器部分，本章选用了较轻量级的分割头，最终利用多层感知机得到语义分割的输出结果。

简单的逐元素相乘来实现空间注意力机制[31]。本文发现，对于空间信息的编码，这样一种简单的构建空间注意力的方式比标准的卷积以及自注意力机制都更加高效。对于解码器，本章的方法从网络中不同的阶段收集多尺度的特征，并使用 Hamburger[134]解码器来进一步提取全局的上下文信息。在此基础上，本章的方法能够（1）从局部扩展到全局，获取多尺度的上下文信息；（2）很好地适配特征的空间与通道维度；（3）将底层级与高层级的信息进行聚合提取。

本章提出的网络称为 SegNeXt，除了解码器部分，其主要由卷积运算组成。而对于解码器，其中包含一个基于分解的 Hamburger 模块[134]（Ham）用于进行全局信息提取。特殊的解码器运用使得本章提出的的 SegNeXt 网络比前有的严重依赖 Transformer 的分割方法更加高效。如图 4.1所示，SegNeXt 显然优于近期的基于 Transformer 网络的方法。尤其在于当 SegNeXt-S 在处理 Cityscapes 数据集中的高分辨率城市场景时，仅仅使用大约 1/6 的计算成本（124.6G vs. 717.1G）与 1/2 的参数（13.9M vs. 27.6M），其效果就超过了 SegFormer-B2（81.3% vs. 81.0%）。本章的贡献可概括为以下几点:

- 本章确定了一个好的语义分割模型应该拥有的特征，并且提出了一个新颖的、专为语义分割任务服务的网络架构，称为 SegNeXt，其利用多尺度的卷积特征来实现空间注意力机制。
- 本章表明仅仅使用简单的卷积构建编码器仍然能够比视觉 Transformer 取得更优的效果，当处理物体细节时，卷积所需的计算资源消耗远小于 Transformer。
- 本章所提出的方法 SegNeXt 在各种分割基准（包括 ADE20K、Cityscapes、COCO-Stuff、Pascal VOC、Pascal Context 和 iSAID 数据集）上远远领先于目前最先进的语义分割方法。



(a) MSCAN 中的核心组成模块　　(b) MSCA

图 4.3　本章所提出的的 MSCA 和 MSCAN 的图示。图中，d,$k_1 \times k_2$ 代表扩张率为 d、核大小为 $k_1 \times k_2$ 的卷积。本章使用不同核大小的卷积提取多尺度特征，然后利用它们作为注意力权重来重新权衡 MSCA 的输入。

第二节　基于卷积自注意力机制的语义分割

SegNeXt 整体的网络架构请参照图 4.2。从模型的整体架构出发来看，SegNeXt 与前有的大多数工作同样采用了编码器-解码器的架构，简单高效。

4.2.1　卷积编码器

SegNeXt 中编码器采用了金字塔结构，这与之前的多数工作[65,70,72]一致。对于编码器中的模块，本文总体采用了与 ViT[5,72] 类似的结构，但不同的是，本文没有使用自注意力机制，而是设计了一个全新的多尺度卷积注意力（MSCA）模块。如图 4.3（a）所示，MSCA 主要由三个部分组成：（1）一个深度卷积用于聚合局部信息；（2）多分支深度卷积用于捕获多尺度上下文信息；（3）一个 1×1 逐点卷积用于模拟特征中不同通道之间的关系。1×1 逐点卷积的输出被直接用作卷积注意力的权重，以重新权衡 MSCA 的输入。本章提出的 MSCA 可以写成如下形式：

$$\mathrm{Att} = \mathrm{Conv}_{1\times1}\left(\sum_{i=0}^{3} \mathrm{Scale}_i(\text{DW-Conv}(F))\right), \tag{4.1}$$

$$\mathrm{Out} = \mathrm{Att} \otimes F, \tag{4.2}$$

其中 F 代表输入特征，Att 和 Out 分别为注意力权重和输出，$\otimes$ 表示逐元素的矩阵乘法运算，DW-Conv 表示深度卷积，$\mathrm{Scale}_i\ (i \in \{0,1,2,3\})$ 表示图 4.3（b）中

表 4.2 不同模型大小的 SegNeXt 网络的详细参数配置。表中，“e.r.” 代表前馈网络的扩展率. “*C*” 和 “*L*” 分别表示通道数和模块的数量 “解码器维度” 表示解码器中 MLP 的维度。“参数量” 是在 ADE20K 数据集[92]上计算的。由于不同的数据集中类别的数量不同，模型的参数量也可能略有变化。

阶段	输出分辨率	e.r.	SegNeXt-T	SegNeXt-S	SegNeXt-B	SegNeXt-L
1	$\frac{H}{4} \times \frac{W}{4} \times C$	8	$C=32, L=3$	$C=64, L=2$	$C=64, L=3$	$C=64, L=3$
2	$\frac{H}{8} \times \frac{W}{8} \times C$	8	$C=64, L=3$	$C=128, L=2$	$C=128, L=3$	$C=128 , L=5$
3	$\frac{H}{16} \times \frac{W}{16} \times C$	4	$C=160, L=5$	$C=320, L=4$	$C=320, L=12$	$C=320, L=27$
4	$\frac{H}{32} \times \frac{W}{32} \times C$	4	$C=256, L=2$	$C=512, L=2$	$C=512, L=3$	$C=512, L=3$
解码器维度			256	256	512	1,024
参数量 (M)			4.3	13.9	27.6	48.9

的第 i 个分支，Scale_0 为残差连接。遵循[130]，在 MSCA 的每个分支中，SegNeXt 使用两个深度条带卷积来近似模拟大卷积核的深度卷积。每个分支的卷积核大小分别被设定为 7、11 和 21。本文选择深度条带卷积主要考虑到以下两方面原因：一方面，相较于普通卷积，条带卷积更加轻量化。为了模拟核大小为 7×7 的标准二维卷积，只需使用一对 7×1 和 1×7 的条带卷积。另一方面，在实际的分割场景中存在一些条状物体，例如人和电线杆。因此，条状卷积可以作为标准网格状的卷积的补充，有助于提取条状特征[130,135]。

将一连串的图中模块堆叠在一起，就能够得到本章提出的卷积编码器，其被命名为 MSCAN。MSCAN 采用一个很普遍的的层级式结构，其包含四个阶段，每个阶段的特征图分辨率递减，分别为 $\frac{H}{4} \times \frac{W}{4}$、$\frac{H}{8} \times \frac{W}{8}$、$\frac{H}{16} \times \frac{W}{16}$ 和 $\frac{H}{32} \times \frac{W}{32}$。这里，$H$ 和 W 分别是原输入图像的高度和宽度。每个阶段都包含一个下采样模块以及一系列上述的模块。下采样模块有一个具有步长为 2、核大小 3×3 的卷积，紧接着是一个批归一化层[110]。值得注意的是，在 MSCAN 的每个模块中，本章使用的是批归一化而不是层归一化，主要原因是在 SegNeXt 中发现批归一化对分割性能的增益会更多。

本章根据不同的模型大小，设计了四个不同规模的编码器模型，分别命名为 MSCAN-T、MSCAN-S、MSCAN-B 和 MSCAN-L。相应的分割模型分别命名为 SegNeXt-T、SegNeXt-S、SegNeXt-B 和 SegNeXt-L。不同规模的分割网络详细参数配置请参照表 4.2。

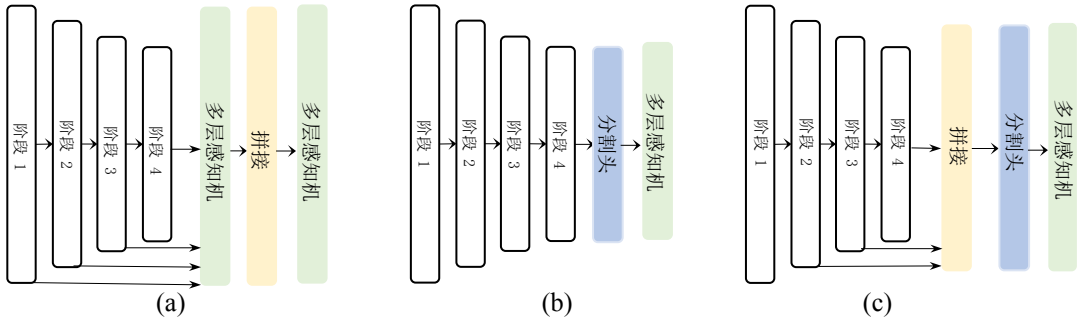


图 4.4 三种不同的解码器网络设计方式。

4.2.2 解码器

在分割模型[70-72]中，编码器大多是在 ImageNet 数据集上预训练的。为了更进一步捕捉高层次的语义，通常需要一个解码器，它被应用在编码器之后。本小节大致概括了前有工作常用的三种简单的解码器结构，如图 4.4所示。第一种在 SegFormer[72]中采用，解码器完全基于一个全局的多层感知机。第二种是主要采用基于 CNN 的模型，在例如 ASPP[70]、PSP[64] 和 DANet[65]中使用。在该结构中，编码器的输出被直接用作解码器分割头的输入。此结构中解码器分割头一般包含更多的参数量与计算量，以此来构建更加精细的分割特征。最后一种是本章的 SegNeXt 采用的结构。SegNeXt 利用一个轻量级的 Hamburger[134]分割头来讲后三个阶段的特征聚合起来，达到进一步对全局上下文信息进行建模的效果。结合强大的卷积编码器，本节发现使用一个轻量级的解码器可以同时提高性能与计算效率。

值得注意的是，SegFormer 的解码器聚合了第 1 阶段到第 4 阶段的特征，而与 SegFormer 不同的是，本章的解码器只使用了最后三个阶段的特征。主要原因在于 SegNeXt 是基于卷积的，第 1 阶段的特征包含了太多的图像的低层级信息，会影响性能。除此以外，由于第 1 阶段的输出特征图分辨率较高，使用其特征图将会带来沉重的计算开销。实验部分将展示本章基于卷积的 SegNeXt 相较于最近的基于 Transformer 的 SegFormer[72]以及 HRFormer[73]在多个数据集上表现得更好。

第三节 实验对比以及结果分析

数据集。 本节在七个流行的数据集上评估了本章提出的方法，包括 ImageNet-1K[99]、ADE20K[92]、Cityscapes[93]、Pascal VOC[131]、Pascal Context[145]、COCO

表 4.3 在 ImageNet-1k 验证集上与最先进方法的比较结果。表中“准确率”表示 Top-1 准确率。

方法	参数量 (M)	准确率 (%)
MiT-B0[72]	3.7	70.5
VAN-Tiny[31]	4.1	75.4
MSCAN-T	4.2	**75.9**
MiT-B1[72]	14.0	78.7
VAN-Small[31]	13.9	81.1
MSCAN-S	14.0	**81.2**
MiT-B2[72]	25.4	81.6
Swin-T[6]	28.3	81.3
ConvNeXt-T[9]	28.6	82.1
VAN-Base[31]	26.6	82.8
MSCAN-B	26.8	**83.0**
MiT-B3[3]	45.2	83.1
Swin-S[6]	49.6	83.0
ConvNeXt-S[6]	50.1	83.1
VAN-Large[31]	44.8	**83.9**
MSCAN-L	45.2	**83.9**

表 4.4 在遥感数据集 iSAID 上与最新方法的比较结果。表中默认采用的是单一尺度（SS）测试。本章提出的 SegNeXt-T 相较于其他方法已经取得了最优的性能。

方法	骨干网络	mIoU (%)
DenseASPP[136]	ResNet50	57.3
PSPNet[64]	ResNet50	60.3
SemanticFPN[137]	ResNet50	62.1
RefineNet[138]	ResNet50	60.2
HRNet[67]	HRNetW-18	61.5
GSCNN[139]	ResNet50	63.4
SFNet[140]	ResNet50	64.3
RANet[141]	ResNet50	62.1
PointRend[142]	ResNet50	62.8
FarSeg[143]	ResNet50	63.7
UperNet[126]	Swin-T	64.6
PointFlow[144]	ResNet50	66.9
SegNeXt-T	MSCAN-T	68.3
SegNeXt-S	MSCAN-S	68.8
SegNeXt-B	MSCAN-B	69.9
SegNeXt-L	MSCAN-L	**70.3**

Stuff[146]，以及 iSAID[147]。ImageNet[99]是最著名的图像分类数据集，其中图片大多以物体为图像中心，共包含 1000 个类别。与大多数分割方法类似，本文使用它对 MSCAN 编码器进行预训练。ADE20K[92] 是一个十分具有挑战性的语义分割数据集，共包含 150 个语义类别，它由 20,210/2,000/3,352 张图像组成的训练、验证和测试集。Cityscapes[93] 主要关注城市场景，包含 5,000 张高分辨率图像，共有 19 个类别，其中有 2,975/500/1,525 张图像，分别用于训练、验证和测试。Pascal VOC[131] 涵盖 20 个前景类和 1 个背景类，经过数据增广后，它有 10, 582/1, 449/1, 456 个图像，分别用于训练、验证和测试。Pascal Context[145] 包含 59 个前景类和 1 个背景类，其训练集和验证集分别包含 4,996 和 5,104 张图像。COCO-Stuff[146] 也是一个具有挑战性的数据集，它包含 172 个语义类别并且其共有 164k 张图像。iSAID[147] 是一个大规模的航空图像分割基准，它包括 15 个前景类和 1 个背景类，其训练、验证和测试集分别涉及 1,411/458/937 张图像。

实验细节。 本章使用 Jittor[148] 和 PyTorch[114] 进行实验。实现是基于 timm（Apache-2.0）[115]库和 mmsegmentation（Apache-2.0）[149] 库开发的，两个代码库分别用于分类任务和分割任务。本文分割模型的所有编码器都在 ImageNet-1K

表 4.5 在不同基准数据集的训练细节。80K + 80K 代表在 Pascal VOC 训练数据集上进行 80K 次迭代的预训练，然后在其训练验证数据集上进行 80K 轮次微调。600K + 40K 代表在 COCO 数据集上进行了 600K 次迭代的预训练，然后在其训练验证数据集上进行了 40K 轮次微调。

数据集	裁剪大小	批大小	迭代次数
ADE20K[92]	512 × 512	16	160K
Cityscapes[93]	1,024×1,024	8	160K
COCO-Stuff[146]	512 × 512	16	80K
Pascal VOC[131]	512 × 512	16	80K + 80K
Pascal VOC[131] w/ COCO[94]	512 × 512	16	600K + 40K
Pascal Context[145]	480 × 480	16	80K
iSAID[147]	896 × 896	16	160K

数据集[99]上进行了预训练。本节分别采用 Top-1 准确率和 mIoU 作为分类和分割的评价指标。所有模型都是在一台含有 8 张 RTX 3090 GPU 的节点上训练的。

对于 ImageNet 预训练，本文的数据增强方法和训练设置与 DeiT[127]相同。对于分割实验，本文采用了一些常见的数据增强方法，包括随机水平翻转、随机缩放（缩放因子从 0.5 到 2）以及随机剪裁。Cityscapes 数据集由于图像分辨率较大，其批大小被设置为 8，并且仅使用该数据集中提供的精细标签。其他所有数据集的批大小为 16。本章采用 AdamW 优化器[116]对模型进行训练，将初始学习率设置为 0.00006，并采用 “poly” 学习率策略，当前学习率等于基础学习率乘以 $(1 - curr\_iter / max\_iter)^{power}$（其中 $power = 0.9$，$curr\_iter$ 和 max_iter 分别表示当前和总共的迭代次数）。本章对 ADE20K、Cityscapes 和 iSAID 数据集进行了 16 万次迭代训练，对 COCO-Stuff、Pascal VOC 和 Pascal Context 数据集进行了 8 万次迭代。在测试过程中，本章同时使用单尺度（SS）和多尺度（MS）的翻转测试策略进行公平的比较。更多的细节请参见表 4.5。

4.3.1 编码器在 ImageNet 数据集上的性能

使用在 ImageNet 预训练的模型是训练分割模型中的一种常见策略，文献[64,70,72-73,132]均有相关探讨。本节在 ImageNet 数据集上对比 MSCAN 与几个近期流行的基于 CNN 和 Transformer 的分类模型的性能表现，如表 4.3 所示。从结果来看，MSCAN 的性能领先于其他目前最先进的基于卷积神经网络的方法如 ConvNeXt[9]，并且能够超越基于 Transformer 的流行方法，例如 Swin-Transformer[6]和 SegFormer[72]中的编码器 MiT。

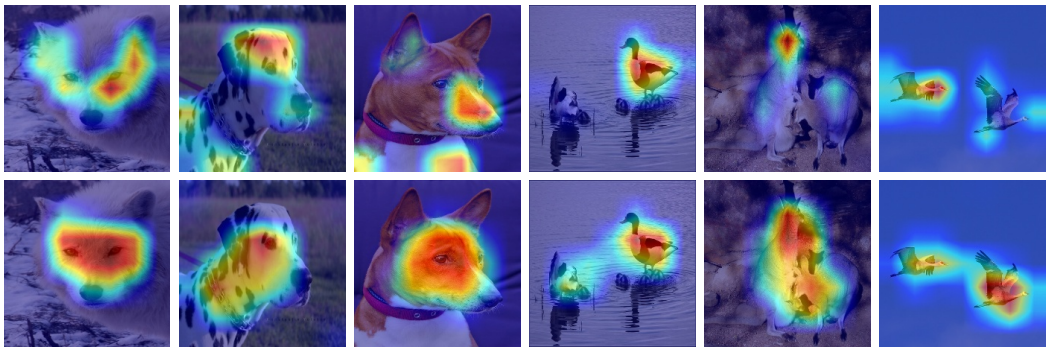


图 4.5 使用 Grad-GAM 的可视化结果。上一行为 ConvNeXt 的可视化结果，下一行为 MSCAN 的结果。

此外，本节还采用 Grad-CAM[150]方法进行可视化研究。如图 4.5 所示，可以明显地看出 MSCAN 产生出更精确的可视化结果，特别是当图像中的物体占据较大区域时（如前三列所示）或当图像中有多个物体时（如后三列所示），ConvNeXt[9]的可视化结果似乎不太准确，而 MSCAN 仍然表现出优秀的结果，这表明了多尺度信息整合和更大感受野的有效性。

表 4.6 关于 MSCA 设计的消融研究。Top-1 表示 ImageNet 数据集上的 Top-1 准确性，mIoU 是在 ADE20K 数据集测试的。Br: 分支。

7 × 7 Br	11 × 11 Br	21 × 21 Br	1 × 1 卷积	卷积注意力	Top-1	mIoU
✓	✗	✗	✓	✓	74.7	39.6
✗	✓	✗	✓	✓	75.2	39.7
✗	✗	✓	✓	✓	75.3	40.0
✓	✓	✓	✗	✓	74.8	39.1
✓	✓	✓	✓	✗	75.5	40.5
✓	✓	✓	✓	✓	75.9	41.1

4.3.2 消融实验

MSCA 设计的消融实验。 本小节在 ImageNet 和 ADE20K 数据集上进行 MSCA 设计的消融研究。K × K 分支包含 1 × K 的深度卷积和 K × 1 的深度卷积。1 × 1 卷积指的是通道混合操作。卷积注意力是指逐元素的乘积，该操作使得网络获得自适应能力。消融结果显示在表 4.6。不难发现，MSCA 中的每个部分都对最终的性能做出了贡献。

表 4.7 解码器中使用不同注意力机制的性能表现。SegNeXt-B w/ Ham 是指 MSCAN-B 编码器加上 Ham 解码器。FLOPs 是用 512×512 的输入尺寸计算的。

结构	参数量 (M)	GFLOPs	mIoU (SS)	mIoU (MS)
SegNeXt-B w/ CC[85]	27.8	35.7	47.3	48.6
SegNeXt-B w/ EMA[87]	27.4	32.3	48.0	49.1
SegNeXt-B w/ NL[28]	27.6	40.9	48.6	50.0
SegNeXt-B w/ Ham[134]	27.6	34.9	48.5	49.9

表 4.8 在解码器中使用不同分割头的性能。表中 SegNeXt-T w/ Ham 表示使用 MSCAN-T 编码器搭配 Ham 解码器。FLOPs 在 512×512 的分辨率下进行计算得到。表中为 COCO-Stuff 数据集上的结果。

结构	参数量 (M)	GFLOPs	mIoU (SS)	mIoU (MS)
SegNeXt-T w/ MSCA	4.4	6.7	38.2	38.6
SegNeXt-T w/ Ham[134]	4.3	6.6	38.7	39.1
SegNeXt-S w/ MSCA	14.0	15.9	42.1	42.4
SegNeXt-S w/ Ham[134]	13.9	15.9	42.2	42.8
SegNeXt-B w/ MSCA	28.0	33.6	45.1	45.5
SegNeXt-B w/ Ham[134]	27.6	34.9	45.8	46.3
SegNeXt-L w/ MSCA	50.1	69.8	45.9	46.4
SegNeXt-L w/ Ham[134]	48.9	70.0	46.5	47.2

解码器的全局上下文信息。 解码器在从多尺度特征中整合全局上下文信息发挥着重要的作用。本小节研究了在解码器中使用不同的全局上下文模块对性能的影响。正如大多数以前的工作[28,65]所示，基于注意力的解码器相较于基于金字塔结构的解码器[64,70]效果更好，因此本小节只展示使用基于注意力的解码器的结果。具体来说，本小节展示了 4 种不同类型的基于注意力的解码器的结果，包括具有 $\mathcal{O}(n^2)$ 复杂度的非局部注意力[28] 和 CCNet[85]，具有 $\mathcal{O}(n)$ 复杂度的 EMANet[87]和 HamNet[134]。如表 4.7所示，Ham 在复杂性和性能之间实现了最佳的权衡。除了使用例如 Ham 这样的基于注意力机制变体的解码器，本小节探索了使用 MSCA 作为本文的分割头。关于两种不同分割头的实验结果请参见表 4.8。从表中不难发现，使用 Ham 的分割头相较于 MSCA 分割头能够取得更好的效果。由此可见，基于 CNN 的编码器更加青睐于拥有全局感受野的分割头。因此，本文在解码器中使用 Hamburger[134]。

表 4.9 不同解码器结构的性能。SegNeXt-T(a) 是指解码器中使用图 4.4中的（a）结构。FLOPs 是用 512×512 的输入尺寸计算的。SegNeXt-T (c) w/ stage 1 代表阶段 1 的输出也被聚合到解码器。

结构	参数量 (M)	GFLOPs	mIoU (SS)	mIoU (MS)
SegNeXt-T (a)	4.4	10.0	40.3	41.1
SegNeXt-T (b)	4.2	4.9	30.9	40.6
SegNeXt-T (c)	4.3	6.6	41.1	42.2
SegNeXt-T (c) w/ stage 1	4.3	12.1	40.7	42.2

表 4.10 本文提出的多尺度卷积注意力机制 (MSCA) 的重要性。SegNeXt-T w/o MSCA 代表仅仅使用单分支的大核卷积[31]来取代 MSCA 的多分支结构。表中 FLOPs 在输入为 512 × 512 的情况下计算得到。

结构	参数量 (M)	GFLOPs	mIoU (SS)	mIoU (MS)
SegNeXt-T w/o MSCA	4.2	6.5	39.5	40.9
SegNeXt-T w/ MSCA	4.3	6.6	41.0	42.5
SegNeXt-S w/o MSCA	13.8	15.8	43.5	45.2
SegNeXt-S w/ MSCA	13.9	15.9	44.3	45.8

解码器结构。 与图像分类任务不同，分割模型需要高分辨率的输出。对于高分辨率的图像而言，计算量是网络设计中必须要考虑的因素之一。本小节为分割任务设计了三种不同形式的解码器，所有这些都已在图 4.4中显示。相应的结果列于表 4.9。可以发现，虽然将阶段 1 的特征一起拼接至分割头中也能取得良好的性能，但是考虑到计算成本，去掉阶段 1 的特征也同样能取得优异的性能，并且计算量更低（如表 4.9中 SegNeXt-T (c) 的结果）。因此，本文选择使用多尺度的特征聚合方式，并且抛弃掉阶段 1 中分辨率较大的特征图以减少计算量。

MSCA 的重要性。 本小节进行实验来证明 MSCA 对于分割的重要性。作为比较，本节跟随前有的 VAN[31] 工作，使用一个大核的单一卷积来取代 MSCA 中的多个分支。实验结果如表 4.10 和表 4.3所示。从表中可以观察到，虽然两个编码器在 ImageNet 分类中的性能接近，但具备多尺度卷积注意力机制的 SegNeXt 相较于不具备多尺度卷积注意力机制能够取得更优越的效果。这表明在语义分割任务中，一个良好的编码器需要聚合多尺度的特征信息，这对最终的结果至关重要。

表 4.11 在 ADE20K、Cityscapes 和 COCO-Stuff 数据集基准上与最先进的方法比较。ADE20K 和 COCO-Stuff 的 FLOPs 数（G）是以 512×512 的输入尺寸计算的，Cityscapes 是以 2,048×1,024 计算的。† 表示在 ImageNet-22K 上预训练的模型。

模型	参数量	ADE20K			Cityscapes			COCO-Stuff		
	(M)	GFLOPs	mIoU (SS/MS)		GFLOPs	mIoU (SS/MS)		GFLOPs	mIoU (SS/MS)	
Segformer-B0[72]	3.8	8.4	37.4	38.0	125.5	76.2	78.1	8.4	35.6	-
SegNeXt-T	4.3	6.6	**41.1**	**42.2**	50.5	**79.8**	**81.4**	6.6	**38.7**	**39.1**
Segformer-B1[72]	13.7	15.9	42.2	43.1	243.7	78.5	80.0	15.9	40.2	-
HRFormer-S[73]	13.5	109.5	44.0	45.1	835.7	80.0	81.0	109.5	37.9	38.9
SegNeXt-S	13.9	15.9	**44.3**	**45.8**	124.6	**81.3**	**82.7**	15.9	**42.2**	**42.8**
Segformer-B2[72]	27.5	62.4	46.5	47.5	717.1	81.0	82.2	62.4	44.6	-
MaskFormer[77]	42	55	46.7	48.8	-	-	-	-	-	-
SegNeXt-B	27.6	34.9	**48.5**	**49.9**	275.7	**82.6**	**83.8**	34.9	**45.8**	**46.3**
SETR-MLA†[71]	310.6	-	48.6	50.1	-	79.3	82.2	-	-	-
DPT-Hybrid[75]	124.0	307.9	-	49.0	-	-	-	-	-	-
Segformer-B3[72]	47.3	79.0	49.4	50.0	962.9	81.7	83.3	79.0	45.5	-
Mask2Former[10]	47	74	47.7	49.6	-	-	-	-	-	-
HRFormer-B[73]	56.2	280.0	48.7	50.0	2223.8	81.9	82.6	280.0	42.4	43.3
MaskFormer[77]	63	79	49.8	51.0	-	-	-	-	-	-
SegNeXt-L	48.9	70.0	**51.0**	**52.1**	577.5	**83.2**	**83.9**	70.0	**46.5**	**47.2**

4.3.3 与当前最先进的方法比较

本小节将 SegNeXt 与最先进的基于 CNN 的方法，如 HRNet[67]、ResNeSt[161]和 EfficientNet[19]，以及基于 Transformer 的方法，如 Swin Transformer[6]、SegFormer[72]、HRFormer[73]、MaskFormer[77]和 Mask2Former[10]进行比较。

性能-计算的权衡。 ADE20K 和 Cityscapes 是语义分割中两个广泛使用的基准。如图4.1所示，本文绘制了 SegNeXt 与前有的方法在 Cityscape 和 ADE20K 验证集上的性能-计算曲线。显然，与其他最先进的方法相比，本文的方法，相较于如 SegFormer[72], HRFormer[73], 和 MaskFormer[77]这些方法，在性能和计算量之间实现了最佳的折衷。

与最先进的 Transformer 比较。特别是，由于 SegFormer[72] 中使用的自注意力机制是平方级的复杂度，而本文的方法使用卷积，这使得 SegNeXt 在处理例如 Cityscapes 这样的高分辨率数据集时不仅仅表现更好，而且效率更高。例如，SegNeXt-B 比 SegFormer-B2 增加了 1.6 mIoU（81.0 v. 82.6），但使用的计算量减少了 40%。在图 4.6与图 4.7中，本章还展示了在 ADE20K 和 CityScapes 数据集上与 SegFormer 的定性比较。不难发现，当语义类别更多时（即在 ADE20K 数据集），SegNeXt 展现出了更好辨别能力。除此以外，由于使用了多尺度的卷积注意力机制，SegNeXt 更加擅长处理物体的细节部分。目前，基于 Transformer

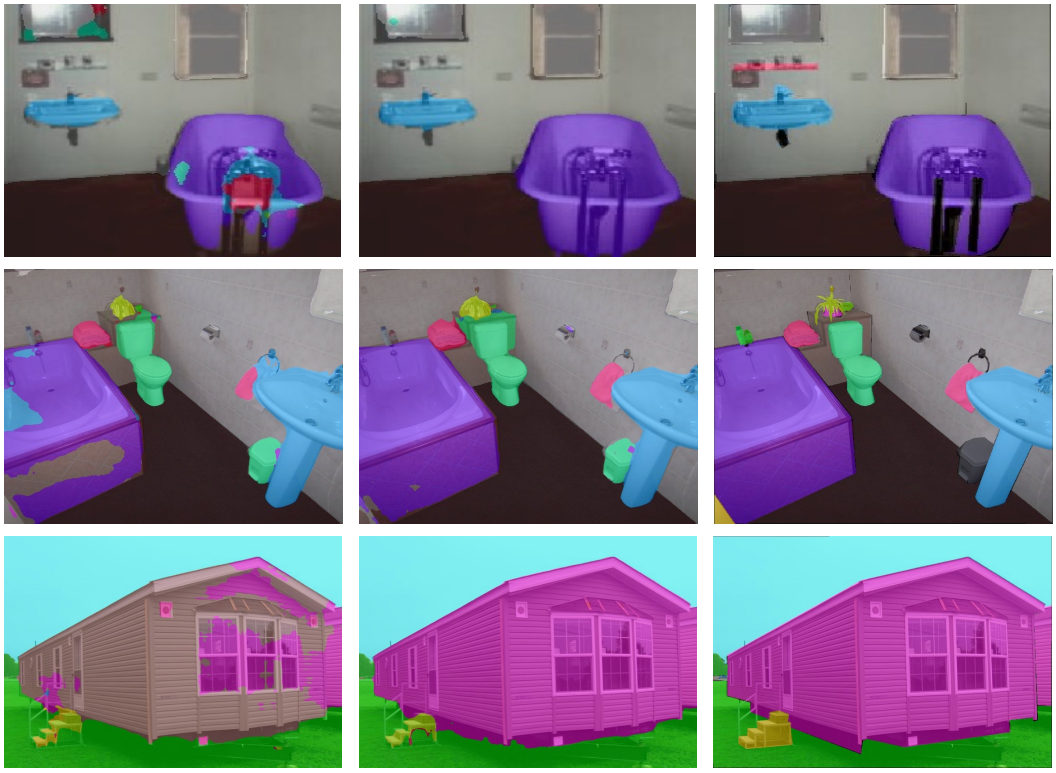


图 4.6 SegNeXt-B 和 SegFormer-B2 在 ADE20K 数据集上的定性比较。左：SegFromer-B2，中：SegNeXt-B，右: 真实数据。

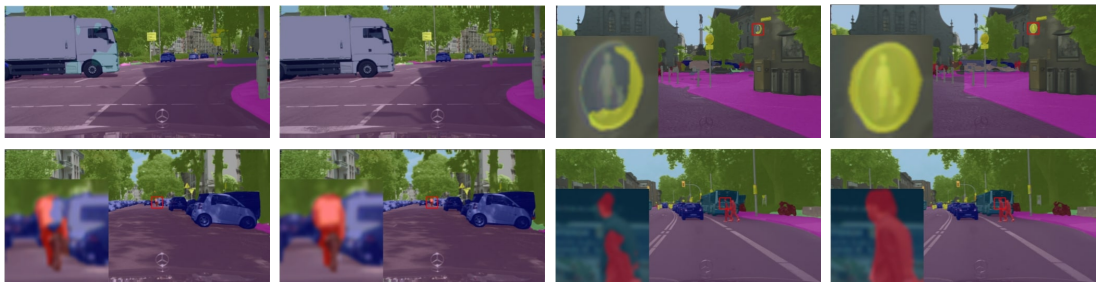
的方法已经成为计算机视觉任务中的主流方法之一。本节则选用了 ADE20K、Cityscapes、COCO-Stuff 和 Pascal Context 基准数据集来比较 SegNeXt 与最先进的基于 Transformer 模型之间的性能差异。实验结果如表 4.11所示。在 ADE20K 数据集上，SegNeXt-L 在参数和计算成本相似的情况下，以 3.3% 的 mIoU（即平均交并比）优势超过了以 Swin-T 为骨干网络的 Mask2Former。此外，与 SegFormer-B2 相比，SegNeXt-B 在 ADE20K 数据集上仅使用其 56% 的计算量就获得了 2.0% 的 mIoU 提升（即 49.5% vs. 46.5%）。值得一提的是，由于 SegFormer 中使用的自注意力机制是平方级复杂度的，而本文采用的是卷积注意力机制，因此 SegNeXt 不仅在处理例如 Cityscapes 这样的高分辨率数据集时表现更好，而且效率更高。例如，SegNeXt-B 相较于 SegFormer-B2 增加了 1.6%mIoU（即 82.6% vs. 81.0%），使用的计算量却减少了 40%。在图 4.6与图 4.7中，本章还展示了在 ADE20K 和 CityScapes 数据集上与 SegFormer 的定性比较结果。可以看出，在面对更多语义类别（即在 ADE20K 数据集）时，SegNeXt 表现出了更好的辨别能力。除此以外，由于采用了多尺度的卷积注意力机制，SegNeXt 能够更加擅长于处理物体的

表 4.12 在 Pascal VOC 数据集上与最先进的方法比较结果。* 表示 COCO[94]预训练。† 表示利用 JFT-300M 数据集[151] 预训练。$ 表示利用额外的 3 亿张无标签图像进行预训练。

方法	骨干网络	mIoU
DANet[65]	ResNet101	82.6
OCRNet[66]	HRNetV2-W48	84.5
HamNet[134]	ResNet101	85.9
EncNet*[90]	ResNet101	85.9
EMANet*[87]	ResNet101	87.7
DeepLabV3+*[133]	Xception-71	87.8
DeepLabV3+†[133]	Xception-JFT	89.0
NAS-FPN$[152]	EfficientNet-L2	90.5
SegNeXt-T	MSCAN-T	82.7
SegNeXt-S	MSCAN-S	85.3
SegNeXt-B	MSCAN-B	87.5
SegNeXt-L*	MSCAN-L	**90.6**

表 4.13 与最先进的实时方法在 Cityscapes 验证数据集上的比较结果。本文用单个 RTX-3090 GPU 和 AMD EPYC 7543 32 核 CPU 处理器对 SegNeXt 进行测试。在不使用任何优化的情况下，SegNeXt-T 可以达到每秒 25 帧（FPS），这达到了实时应用的要求。

方法	输入分辨率	mIoU
ESPNet[153]	512×1,024	60.3
ESPNetv2[154]	512×1,024	66.2
ICNet[155]	1,024 × 2,048	69.5
DFANet[156]	1,024 × 1,024	71.3
BiSeNet[157]	768 × 1,536	74.6
BiSeNetv2[158]	512 × 1,024	75.3
DF2-Seg[159]	1,024 × 2,048	74.8
SwiftNet[160]	1,024 × 2,048	75.5
SFNet[140]	1,024 × 2,048	77.8
SegNeXt-T	768 × 1,536	**78.0**



SegFormer-B2 SegNeXt-B SegFormer-B2 SegNeXt-B

图 4.7 SegNeXt-B 和 SegFormer-B2 在 Cityscapes 数据集上的定性比较。

细节部分。

与最先进的 CNN 方法比较。表 4.4、表 4.12、和表 4.14展示了在 Pascal VOC 2012、Pascal Context 和 iSAID 数据集上 SegNeXt 与最先进的 CNN 方法的比较结果，包括的方法例如 ResNeSt-269[161]、EfficientNet-L2[152]和 HRNetW48[67]。SegNeXt-L 的性能优于专为分割任务设计的 HRNet（OCR）[66-67] 模型（60.3% v.56.3%），其使用的参数和计算量甚至更少。此外，SegNeXt-L 在 Pascal VOC 2012 在线测试排行榜上的表现甚至优于 EfficientNet-L2（NAS-FPN）方法，后者在额外的 3 亿张不可获取的图像上进行了预训练。值得注意的是，EfficientNet-L2（NAS-FPN）拥有惊人的 485M 的参数，而 SegNeXt-L 只有 48.7M 的参数。

与实时方法的比较。除了最先进的性能外，本章的方法也适合于实时部署。

表 4.14 在 Pascal Context 基准数据集上的比较结果。FLOPs 是以 512×512 的输入尺寸计算得到的。* 表示 ImageNet-22K 预训练。† 表示 ADE20K 预训练。

方法	骨干网络	参数量 (M)	GFLOPs	mIoU (SS/MS)	
PSPNet[64]	ResNet101	-	-	-	47.8
DANet[65]	ResNet101	69.1	277.7	-	52.6
EMANet[87]	ResNet101	61.1	246.1	-	53.1
HamNet[134]	ResNet101	69.1	277.9	-	55.2
HRNet(OCR)[67]	HRNetW48	74.5	-	-	56.2
DeepLabV3+[133]	ResNeSt-269	-	-	-	58.9
SETR-PUP*[71]	ViT-Large	317.8	-	54.4	55.3
SETR-MLA*[71]	ViT-Large	309.5	-	54.9	55.8
HRFormer-B[73]	HRFormer-B	56.2	280.0	57.6	58.5
DPT-Hybrid†[75]	ViT-Hybrid	124.0	-	-	60.5
SegNeXt-T	MSCAN-T	4.2	6.6	51.2	53.3
SegNeXt-S	MSCAN-S	13.9	15.9	54.2	56.1
SegNeXt-B	MSCAN-B	27.6	34.9	57.0	59.0
SegNeXt-L	MSCAN-L	48.8	70.0	58.7	60.3
SegNeXt-L†	MSCAN-L	48.8	70.0	**59.2**	**60.9**

即使没有任何特定的软件或硬件加速，SegNeXt-T 在处理 768×1,536 尺寸的图像时，使用单个 3090 RTX GPU 实现了每秒 25 帧（FPS）。如表 4.13所示，文章的方法为 Cityscapes 验证集的实时分割创造了新的最领先的结果。

第四节 本章小结

本章首先分析了先前成功的分割模型，并找到它们所拥有的良好属性。基于这些发现，本章提出了一个定制化的、多尺度的卷积注意模块（MSCA）和一个完全基于卷积神经网络的网络，称为 SegNeXt。实验结果表明，SegNeXt 在多个基准数据集上以相当大的领先优势上超过了目前最先进的基于 Transformer 的方法。

近年来，基于 Transformer 的模型在各种视觉任务排行榜上占主导地位。相反，本章显示，当使用适当的设计时，基于卷积神经网络的方法仍然可以比基于 Transformer 的方法表现得更好。希望本章能够鼓励研究人员进一步研究基于 CNN 方法的潜力。与此同时，本章所提出的模型也有其局限性。例如，本章并没有探索将该方法扩展到具有 100M 以上参数量的大规模模型以及在其他视觉或自然语言处理任务上的表现。这些局限性将在未来的工作中解决。值得注意

的是，本章提出的 MSCA 模块可以以不显著增加计算代价的方式轻松集成到任何基于卷积神经网络的体系结构中。因此，本章所提出的 MSCA 模块是可扩展的，可用于解决各种图像分割问题。另外，未来的工作还可以探索 MSCA 模块在其他视觉任务中的效果，例如人脸识别和目标检测。

第五章 总结与展望

第一节 文章总结

骨干网络一直以来是计算机视觉领域研究热点领域之一，其负责从输入数据中提取视觉特征，一般来说骨干网络会在较大的数据集（例如 ImageNet-1k）上进行预训练，随后在特定下游任务中进行微调。现有的一些工作[5-6]使用以自注意力为基础的 Transformer 骨干网络架构。然而，这些工作一般需要大量数据进行预训练，模型难以收敛，举例来说，ViT[5]在提出时使用谷歌私有的 JFT-300M 数据集进行预训练才能够很好的收敛。同时，自注意力本身因其计算量与输入序列成平方关系，训练时对计算资源要求较高。为了解决上述问题，本文设计了卷积注意力机制，旨在简化自注意力机制，以纯卷积架构构建骨干网络。卷积注意力机制能够充分利用大核卷积的优势，并且在卷积神经网络中引入一定的自适应性。本文的主要研究内容和贡献可以总结如下：

（1）本文总结分析了现有骨干网络中存在的一些缺点，在此基础上设计了卷积注意力机制，并以此为基础模块构建层级式的骨干网络。卷积注意力机制以大核卷积为核心，利用大核卷积生成权重，使用 Hadamard 积对输入特征图进行重新加权变换。通过使用 Hadamard 积，Conv2Former 能够一定程度上根据输入图片自动调整注意力权重，赋予其自适应的能力。同时，本文探索了在卷积注意力基础下使用更大的卷积核，发现模型性能能够随着卷积核的增大而提升。当将 Conv2Former 修改为各向同性的架构时，本文的 Conv2Former 在性能表现上同样具有很强的竞争力。

（2）本文在多个基准数据集上将 Conv2Former 与其他一些比较先进的方法进行对比，包括 ImageNet-1k、COCO、ADE20K 三个数据集。同时，本文对得到的实验结果进行了一定的分析，Conv2Former 在多个数据集上的结果均领先于当前最先进的方法。同时，本文对 Conv2Former 中的 Hadamard 积融合策略进行了一些消融实验，包括使用 Softmax 函数、L1 归一化方式等等，结果表明直接使用 Hadamard 积效果最优。

（3）基于卷积注意力的设计，本文为特定的下游任务，即语义分割任务，定

制了一个骨干网络并搭建语义分割网络，称为 SegNeXt。本文分析了前有的成功的语义分割网络中具有的一些特性，在卷积注意力机制中引入多尺度信息并且在模型的宏观设计上进行了一些改变以使得模型适配语义分割任务。

（4）本文在多个语义分割的基准数据集上达到了最先进的效果，性能超越了 MaskFormer[77]、HRFormer[73]等当前最先进的方法，并且对实验结果进行了可视化的分析。同时本文对多尺度卷积注意力中的每个分支进行了消融实验，表明多尺度中每个分支的必要性。

第二节 未来展望

本文为网络架构设计提供了新的思路，即卷积注意力机制，并且在多个基准数据集上取得了非常优秀的性能。然而，卷积注意力作为一个全新的架构，仍然有很多可以提升的空间。本文为未来可探索的方向提供如下两点思路：

（1）卷积注意力是全新的网络架构形式，其潜力有待发掘。当下比较流行的卷积神经网络与 Transformer 的混合模型普遍性能比较高，而配合卷积注意力机制的混合模型在本文并没有充分探索。未来能够探索将卷积注意力与自注意力结合的骨干网络架构。

（2）目前由于 Pytorch 对深度卷积的优化存在缺陷，本文的卷积注意力比较依赖于深度卷积导致实际推理速度令人堪忧。对卷积注意力的加速推理同样是未来值得探索的方向之一。

参考文献

[1] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems, 2012, 25: 1097-1105.

[2] HE K, CHEN X, XIE S, et al. Masked Autoencoders Are Scalable Vision Learners[Z]. 2021. arXiv: 2111.06377 [cs.CV].

[3] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 770-778.

[4] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 4700-4708.

[5] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[C]//International Conference on Learning Representations. 2020.

[6] LIU Z, LIN Y, CAO Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]//IEEE/CVF International Conference on Computer Vision. 2021: 10012-10022.

[7] TOLSTIKHIN I O, HOULSBY N, KOLESNIKOV A, et al. Mlp-mixer: An all-mlp architecture for vision[J]. Advances in neural information processing systems, 2021, 34: 24261-24272.

[8] YU W, LUO M, ZHOU P, et al. MetaFormer is Actually What You Need for Vision[J]. arXiv preprint arXiv:2111.11418, 2021.

[9] LIU Z, MAO H, WU C Y, et al. A convnet for the 2020s[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 11976-11986.

[10] CHENG B, MISRA I, SCHWING A G, et al. Masked-attention mask transformer for universal image segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 1290-1299.

[11] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[12] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[13] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the inception architecture for computer vision[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 2818-2826.

[14] SZEGEDY C, IOFFE S, VANHOUCKE V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]//Association for the Advancement of Artificial Intelligence. 2017.

[15] XIE S, GIRSHICK R, DOLLÁR P, et al. Aggregated residual transformations for deep neural networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 1492-1500.

[16] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 7132-7141.

[17] WOO S, PARK J, LEE J Y, et al. Cbam: Convolutional block attention module[C]//European Conference on Computer Vision. 2018: 3-19.

[18] HOU Q, ZHOU D, FENG J. Coordinate attention for efficient mobile network design[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 13713-13722.

[19] TAN M, LE Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]//International conference on machine learning. 2019: 6105-6114.

[20] TAN M, LE Q. Efficientnetv2: Smaller models and faster training[C]//International Conference on Machine Learning. 2021: 10096-10106.

[21] HOWARD A, SANDLER M, CHU G, et al. Searching for mobilenetv3[C]//IEEE/CVF International Conference on Computer Vision. 2019: 1314-1324.

[22] ZOPH B, LE Q V. Neural architecture search with reinforcement learning[J]. arXiv preprint arXiv:1611.01578, 2016.

[23] HUANG Y, CHENG Y, BAPNA A, et al. Gpipe: Efficient training of giant neural networks using pipeline parallelism[J]. Advances in Neural Information Processing Systems, 2019, 32: 103-112.

[24] TOUVRON H, VEDALDI A, DOUZE M, et al. Fixing the train-test resolution discrepancy[J]. arXiv preprint arXiv:1906.06423, 2019.

[25] XIE C, TAN M, GONG B, et al. Adversarial examples improve image recognition[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 819-828.

[26] BROCK A, DE S, SMITH S L, et al. High-Performance Large-Scale Image Recognition Without Normalization[J]. arXiv preprint arXiv:2102.06171, 2021.

[27] RADOSAVOVIC I, KOSARAJU R P, GIRSHICK R, et al. Designing network design spaces[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 10428-10436.

[28] WANG X, GIRSHICK R, GUPTA A, et al. Non-local neural networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 7794-7803.

[29] DING X, ZHANG X, HAN J, et al. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 11963-11975.

[30] RAO Y, ZHAO W, TANG Y, et al. HorNet: Efficient High-Order Spatial Interactions with Recursive Gated Convolutions[J]. arXiv preprint arXiv:2207.14284, 2022.

[31] GUO M H, LU C Z, LIU Z N, et al. Visual Attention Network[J]. arXiv preprint arXiv:2202.09741, 2022.

[32] LIU S, CHEN T, CHEN X, et al. More ConvNets in the 2020s: Scaling up Kernels Beyond 51x51 using Sparsity[J]. arXiv preprint arXiv:2207.03620, 2022.

[33] DAI J, QI H, XIONG Y, et al. Deformable convolutional networks[C]//IEEE/CVF International Conference on Computer Vision. 2017: 764-773.

[34] MNIH V, HEESS N, GRAVES A, et al. Recurrent models of visual attention[C]// Advances in Neural Information Processing Systems. 2014: 2204-2212.

[35] GUO M H, CAI J X, LIU Z N, et al. Pct: Point cloud transformer[J]. Computational Visual Media, 2021, 7(2): 187-199.

[36] CHEN L, ZHANG H, XIAO J, et al. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 5659-5667.

[37] WANG Q, WU B, ZHU P, et al. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks[Z]. 2020. arXiv: 1910.03151 [cs.CV].

[38] TOUVRON H, CORD M, DOUZE M, et al. Training data-efficient image transformers & distillation through attention[J]. arXiv preprint arXiv:2012.12877, 2020.

[39] HEO B, YUN S, HAN D, et al. Rethinking Spatial Dimensions of Vision Transformers [J]. arXiv preprint arXiv:2103.16302, 2021.

[40] WANG W, XIE E, LI X, et al. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions[C]//IEEE/CVF International Conference on Computer Vision. 2021: 568-578.

[41] YANG J, LI C, ZHANG P, et al. Focal attention for long-range interactions in vision transformers[J]. Advances in Neural Information Processing Systems, 2021, 34: 30008-30022.

[42] DONG X, BAO J, CHEN D, et al. Cswin transformer: A general vision transformer backbone with cross-shaped windows[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 12124-12134.

[43] LI Y, WU C Y, FAN H, et al. MViTv2: Improved Multiscale Vision Transformers for Classification and Detection[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 4804-4814.

[44] CHEN C F, FAN Q, PANDA R. CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification[J]. arXiv preprint arXiv:2103.14899, 2021.

[45] HAN K, XIAO A, WU E, et al. Transformer in transformer[J]. arXiv preprint arXiv:2103.00112, 2021.

[46] WU H, XIAO B, CODELLA N, et al. Cvt: Introducing convolutions to vision transformers[C]//IEEE/CVF International Conference on Computer Vision. 2021: 22-31.

[47] VASWANI A, RAMACHANDRAN P, SRINIVAS A, et al. Scaling Local Self-Attention For Parameter Efficient Visual Backbones[J]. arXiv preprint arXiv:2103.12731, 2021.

[48] GUO J, HAN K, WU H, et al. Cmt: Convolutional neural networks meet vision transformers[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 12175-12185.

[49] DAI Z, LIU H, LE Q, et al. Coatnet: Marrying convolution and attention for all data sizes[J]. Advances in Neural Information Processing Systems, 2021, 34.

[50] YUAN L, CHEN Y, WANG T, et al. Tokens-to-Token ViT: Training Vision Transformers From Scratch on ImageNet[C]//IEEE/CVF International Conference on Computer Vision. 2021: 558-567.

[51] HAN Q, FAN Z, DAI Q, et al. Demystifying local vision transformer: Sparse connectivity, weight sharing, and dynamic weight[J]. arXiv preprint arXiv:2106.04263, 2021.

[52] HASSANI A, WALTON S, LI J, et al. Neighborhood Attention Transformer[J]. arXiv preprint arXiv:2204.07143, 2022.

[53] ZHOU D, KANG B, JIN X, et al. Deepvit: Towards deeper vision transformer[J]. arXiv preprint arXiv:2103.11886, 2021.

[54] ZHAI X, KOLESNIKOV A, HOULSBY N, et al. Scaling Vision Transformers[J]. arXiv preprint arXiv:2106.04560, 2021.

[55] TOUVRON H, CORD M, SABLAYROLLES A, et al. Going deeper with image transformers[C]//IEEE/CVF International Conference on Computer Vision. 2021: 32-42.

[56] LIU Z, HU H, LIN Y, et al. Swin transformer v2: Scaling up capacity and resolution[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 12009-12019.

[57] YUAN L, HOU Q, JIANG Z, et al. Volo: Vision outlooker for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022.

[58] JIANG Z H, HOU Q, YUAN L, et al. All tokens matter: Token labeling for training better vision transformers[J]. Advances in Neural Information Processing Systems, 2021, 34: 18590-18602.

[59] BAO H, DONG L, PIAO S, et al. BEiT: BERT Pre-Training of Image Transformers[C]//International Conference on Learning Representations. 2022.

[60] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2015: 3431-3440.

[61] BADRINARAYANAN V, KENDALL A, CIPOLLA R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.

[62] RONNEBERGER O, FISCHER P, BROX T. U-net: Convolutional networks for biomedical image segmentation[C]//International Conference on Medical image computing and computer-assisted intervention. 2015: 234-241.

[63] YU F, KOLTUN V. Multi-scale context aggregation by dilated convolutions[J]. arXiv preprint arXiv:1511.07122, 2015.

[64] ZHAO H, SHI J, QI X, et al. Pyramid scene parsing network[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 2881-2890.

[65] FU J, LIU J, TIAN H, et al. Dual attention network for scene segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 3146-3154.

[66] YUAN Y, CHEN X, WANG J. Object-contextual representations for semantic segmentation[C]//European Conference on Computer Vision. 2020: 173-190.

[67] WANG J, SUN K, CHENG T, et al. Deep high-resolution representation learning for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020.

[68] GAO S H, CHENG M M, ZHAO K, et al. Res2Net: A New Multi-scale Backbone Architecture[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(2): 652-662.

[69] LI X, ZHAO H, HAN L, et al. Gated fully fusion for semantic segmentation[C]// Association for the Advancement of Artificial Intelligence. 2020: 11418-11425.

[70] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(4): 834-848.

[71] ZHENG S, LU J, ZHAO H, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 6881-6890.

[72] XIE E, WANG W, YU Z, et al. SegFormer: Simple and efficient design for semantic segmentation with transformers[J]. Advances in Neural Information Processing Systems, 2021, 34.

[73] YUAN Y, FU R, HUANG L, et al. HRFormer: High-Resolution Vision Transformer for Dense Predict[J]. Advances in Neural Information Processing Systems, 2021, 34.

[74] STRUDEL R, GARCIA R, LAPTEV I, et al. Segmenter: Transformer for semantic segmentation[C]//IEEE/CVF International Conference on Computer Vision. 2021: 7262-7272.

[75] RANFTL R, BOCHKOVSKIY A, KOLTUN V. Vision transformers for dense prediction[C]//IEEE/CVF International Conference on Computer Vision. 2021: 12179-12188.

[76] LI X, ZHANG W, PANG J, et al. Video k-net: A simple, strong, and unified baseline for video segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 18847-18857.

[77] CHENG B, SCHWING A, KIRILLOV A. Per-pixel classification is not all you need for semantic segmentation[J]. Advances in Neural Information Processing Systems, 2021, 34: 17864-17875.

[78] HE T, ZHANG Z, ZHANG H, et al. Bag of tricks for image classification with convolutional neural networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 558-567.

[79] LEE Y, KIM J, WILLETTE J, et al. MPViT: Multi-Path Vision Transformer for Dense Prediction[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[80] ZHEN M, WANG J, ZHOU L, et al. Joint semantic segmentation and boundary detection using iterative pyramid contexts[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 13666-13675.

[81] BERTASIUS G, SHI J, TORRESANI L. Semantic segmentation with boundary neural fields[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 3602-3610.

[82] DING H, JIANG X, LIU A Q, et al. Boundary-aware feature propagation for scene segmentation[C]//IEEE/CVF International Conference on Computer Vision. 2019: 6819-6829.

[83] LI X, LI X, ZHANG L, et al. Improving semantic segmentation via decoupled body and edge supervision[C]//European Conference on Computer Vision. 2020: 435-452.

[84] YUAN Y, XIE J, CHEN X, et al. Segfix: Model-agnostic boundary refinement for segmentation[C]//European Conference on Computer Vision. 2020: 489-506.

[85] HUANG Z, WANG X, HUANG L, et al. Ccnet: Criss-cross attention for semantic segmentation[C]//IEEE/CVF International Conference on Computer Vision. 2019: 603-612.

[86] YUAN Y, HUANG L, GUO J, et al. Ocnet: Object context network for scene parsing [J]. arXiv preprint arXiv:1809.00916, 2018.

[87] LI X, ZHONG Z, WU J, et al. Expectation-maximization attention networks for semantic segmentation[C]//IEEE/CVF International Conference on Computer Vision. 2019: 9167-9176.

[88] GUO M H, LIU Z N, MU T J, et al. Beyond self-attention: External attention using two linear layers for visual tasks[J]. arXiv preprint arXiv:2105.02358, 2021.

[89] HE J, DENG Z, ZHOU L, et al. Adaptive pyramid context network for semantic segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 7519-7528.

[90] ZHANG H, DANA K, SHI J, et al. Context encoding for semantic segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 7151-7160.

[91] CHEN Y, DAI X, CHEN D, et al. Mobile-former: Bridging mobilenet and transformer [C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 5270-5279.

[92] ZHOU B, ZHAO H, PUIG X, et al. Scene parsing through ade20k dataset[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 633-641.

[93] CORDTS M, OMRAN M, RAMOS S, et al. The cityscapes dataset for semantic urban scene understanding[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 3213-3223.

[94] LIN T Y, MAIRE M, BELONGIE S, et al. Microsoft coco: Common objects in context [C]//Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. 2014: 740-755.

[95] RADFORD A, KIM J W, HALLACY C, et al. Learning transferable visual models from natural language supervision[C]//International conference on machine learning. 2021: 8748-8763.

[96] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[97] SANDLER M, HOWARD A, ZHU M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 4510-4520.

[98] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet Classification with Deep Convolutional Neural Networks[C/OL]//PEREIRA F, BURGES C, BOTTOU L, et al. Advances in Neural Information Processing Systems: vol. 25. Curran Associates, Inc., 2012. https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[99] DENG J, DONG W, SOCHER R, et al. Imagenet: A large-scale hierarchical image database[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2009: 248-255.

[100] ZHOU D, HOU Q, CHEN Y, et al. Rethinking bottleneck structure for efficient mobile network design[C]//European Conference on Computer Vision. 2020: 680-697.

[101] BELLO I, ZOPH B, VASWANI A, et al. Attention augmented convolutional networks [C]//IEEE/CVF International Conference on Computer Vision. 2019: 3286-3295.

[102] SRINIVAS A, LIN T Y, PARMAR N, et al. Bottleneck transformers for visual recognition[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 16519-16529.

[103] GUO M H, LU C Z, HOU Q, et al. SegNeXt: Rethinking Convolutional Attention Design for Semantic Segmentation[C]//Advances in Neural Information Processing Systems. 2022.

[104] TAN M, LE Q V. Mixconv: Mixed depthwise convolutional kernels[J]. arXiv preprint arXiv:1907.09595, 2019.

[105] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Advances in Neural Information Processing Systems. 2017: 5998-6008.

[106] CHOLLET F. Xception: Deep learning with depthwise separable convolutions[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 1251-1258.

[107] DING X, ZHANG X, MA N, et al. Repvgg: Making vgg-style convnets great again[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 13733-13742.

[108] YANG J, LI C, GAO J. Focal Modulation Networks[J]. arXiv preprint arXiv:2203.11926, 2022.

[109] BA J L, KIROS J R, HINTON G E. Layer Normalization[Z]. 2016. arXiv: 1607.06450 [stat.ML].

[110] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//International Conference on Machine Learning. 2015: 448-456.

[111] HENDRYCKS D, GIMPEL K. Gaussian error linear units (gelus)[J]. arXiv preprint arXiv:1606.08415, 2016.

[112] YANG C, QIAO S, YU Q, et al. MOAT: Alternating Mobile Convolution and Attention Brings Strong Vision Models[J]. arXiv preprint arXiv:2210.01820, 2022.

[113] CAI H, GAN C, HAN S. EfficientViT: Enhanced Linear Attention for High-Resolution Low-Computation Visual Recognition[J]. arXiv preprint arXiv:2205.14756, 2022.

[114] PASZKE A, GROSS S, MASSA F, et al. Pytorch: An imperative style, high-performance deep learning library[C]//Advances in Neural Information Processing Systems. 2019: 8026-8037.

[115] WIGHTMAN R. PyTorch Image Models[Z]. https://github.com/rwightman/pytorch-image-models(Apache-2.0). 2019.

[116] LOSHCHILOV I, HUTTER F. Decoupled weight decay regularization[J]. arXiv preprint arXiv:1711.05101, 2017.

[117] ZHANG H, CISSE M, DAUPHIN Y N, et al. mixup: Beyond empirical risk minimization[J]. arXiv preprint arXiv:1710.09412, 2017.

[118] YUN S, HAN D, OH S J, et al. Cutmix: Regularization strategy to train strong classifiers with localizable features[C]//IEEE/CVF International Conference on Computer Vision. 2019: 6023-6032.

[119] HUANG G, SUN Y, LIU Z, et al. Deep networks with stochastic depth[C]//European Conference on Computer Vision. 2016: 646-661.

[120] ZHONG Z, ZHENG L, KANG G, et al. Random erasing data augmentation[C]//Association for the Advancement of Artificial Intelligence. 2020: 13001-13008.

[121] CUBUK E D, ZOPH B, SHLENS J, et al. Randaugment: Practical automated data augmentation with a reduced search space[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop. 2020: 702-703.

[122] HE K, GKIOXARI G, DOLLAR P, et al. Mask R-CNN[C]//IEEE/CVF International Conference on Computer Vision. 2017.

[123] CAI Z, VASCONCELOS N. Cascade r-cnn: High quality object detection and instance segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019.

[124] REZATOFIGHI H, TSOI N, GWAK J, et al. Generalized intersection over union: A metric and a loss for bounding box regression[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 658-666.

[125] CHEN K, WANG J, PANG J, et al. MMDetection: Open mmlab detection toolbox and benchmark[J]. arXiv preprint arXiv:1906.07155, 2019.

[126] XIAO T, LIU Y, ZHOU B, et al. Unified perceptual parsing for scene understanding[C]//European Conference on Computer Vision. 2018: 418-434.

[127] TOUVRON H, CORD M, DOUZE M, et al. Training data-efficient image transformers & distillation through attention[C]//International Conference on Machine Learning. 2021: 10347-10357.

[128] MAAZ M, SHAKER A, CHOLAKKAL H, et al. EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications[J]. arXiv preprint arXiv:2206.10589, 2022.

[129] CARION N, MASSA F, SYNNAEVE G, et al. End-to-end object detection with transformers[C]//European Conference on Computer Vision. 2020: 213-229.

[130] PENG C, ZHANG X, YU G, et al. Large kernel matters–improve semantic segmentation by global convolutional network[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 4353-4361.

[131] EVERINGHAM M, VAN GOOL L, WILLIAMS C K, et al. The pascal visual object classes (voc) challenge[J]. International Journal of Computer Vision, 2010, 88(2): 303-338.

[132] CHEN L C, PAPANDREOU G, SCHROFF F, et al. Rethinking atrous convolution for semantic image segmentation[J]. arXiv preprint arXiv:1706.05587, 2017.

[133] CHEN L C, ZHU Y, PAPANDREOU G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation[C]//European Conference on Computer Vision. 2018: 801-818.

[134] GENG Z, GUO M H, CHEN H, et al. Is Attention Better Than Matrix Decomposition? [C]//International Conference on Learning Representations. 2021.

[135] HOU Q, ZHANG L, CHENG M M, et al. Strip pooling: Rethinking spatial pooling for scene parsing[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 4003-4012.

[136] YANG M, YU K, ZHANG C, et al. Denseaspp for semantic segmentation in street scenes[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 3684-3692.

[137] KIRILLOV A, GIRSHICK R, HE K, et al. Panoptic feature pyramid networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 6399-6408.

[138] LIN G, MILAN A, SHEN C, et al. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 1925-1934.

[139] TAKIKAWA T, ACUNA D, JAMPANI V, et al. Gated-scnn: Gated shape cnns for semantic segmentation[C]//IEEE/CVF International Conference on Computer Vision. 2019: 5229-5238.

[140] LI X, YOU A, ZHU Z, et al. Semantic flow for fast and accurate scene parsing[C]//European Conference on Computer Vision. 2020: 775-793.

[141] MOU L, HUA Y, ZHU X X. A relation-augmented fully convolutional network for semantic segmentation in aerial scenes[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 12416-12425.

[142] KIRILLOV A, WU Y, HE K, et al. Pointrend: Image segmentation as rendering[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 9799-9808.

[143] ZHENG Z, ZHONG Y, WANG J, et al. Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 4096-4105.

[144] LI X, HE H, LI X, et al. Pointflow: Flowing semantics through points for aerial image segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 4217-4226.

[145] MOTTAGHI R, CHEN X, LIU X, et al. The role of context for object detection and semantic segmentation in the wild[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2014: 891-898.

[146] CAESAR H, UIJLINGS J, FERRARI V. Coco-stuff: Thing and stuff classes in context [C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 1209-1218.

[147] WAQAS ZAMIR S, ARORA A, GUPTA A, et al. isaid: A large-scale dataset for instance segmentation in aerial images[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop. 2019: 28-37.

[148] HU S M, LIANG D, YANG G Y, et al. Jittor: a novel deep learning framework with meta-operators and unified graph execution[J]. Science China Information Sciences, 2020, 63(222103): 1-21.

[149] CONTRIBUTORS M. MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark[Z]. https://github.com/open-mmlab/mmsegmentation(Apache-2.0). 2020.

[150] SELVARAJU R R, COGSWELL M, DAS A, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization[C]//IEEE/CVF International Conference on Computer Vision. 2017: 618-626.

[151] SUN C, SHRIVASTAVA A, SINGH S, et al. Revisiting unreasonable effectiveness of data in deep learning era[C]//IEEE/CVF International Conference on Computer Vision. 2017: 843-852.

[152] ZOPH B, GHIASI G, LIN T Y, et al. Rethinking pre-training and self-training[J]. Advances in Neural Information Processing Systems, 2020, 33: 3833-3845.

[153] MEHTA S, RASTEGARI M, CASPI A, et al. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation[C]//European Conference on Computer Vision. 2018: 552-568.

[154] MEHTA S, RASTEGARI M, SHAPIRO L, et al. Espnetv2: A light-weight, power efficient, and general purpose convolutional neural network[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 9190-9200.

[155] ZHAO H, QI X, SHEN X, et al. Icnet for real-time semantic segmentation on high-resolution images[C]//European Conference on Computer Vision. 2018: 405-420.

[156] LI H, XIONG P, FAN H, et al. Dfanet: Deep feature aggregation for real-time semantic segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 9522-9531.

[157] YU C, WANG J, PENG C, et al. Bisenet: Bilateral segmentation network for real-time semantic segmentation[C]//European Conference on Computer Vision. 2018: 325-341.

[158] YU C, GAO C, WANG J, et al. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation[J]. International Journal of Computer Vision, 2021, 129(11): 3051-3068.

[159] LI X, ZHOU Y, PAN Z, et al. Partial order pruning: for best speed/accuracy trade-off in neural architecture search[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 9145-9153.

[160] ORSIC M, KRESO I, BEVANDIC P, et al. In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 12607-12616.

[161] ZHANG H, WU C, ZHANG Z, et al. Resnest: Split-attention networks[J]. arXiv preprint arXiv:2004.08955, 2020.

致谢

时光荏苒，转眼间我的硕士旅程即将结束。在毕业论文即将完成之际，回顾这三年，感慨万千。科研之路不易，在此，向为我提供帮助的老师、同学以及家人表达由衷的感谢。

首先，我要感谢我的导师程明明教授。依稀记得 2019 年的夏天在程老师的带领下我第一次来到媒体计算实验室，程老师详细地介绍了实验室情况，这是我硕士三年科研之路的开始。程老师以身作则，言传身教，用行动督促着我们，同时在整个科研过程中给予了极大的支持和帮助，在我遇到问题时给予了耐心的指导和建议，让我受益匪浅。十分感谢程老师为我提供的宝贵的研究机会。此外，感谢在字节实习时给予我很大帮助的冯佳时老师和靳潇杰老师，两位老师的指导让我终身受益。

实验室的时光让我结识了许多小伙伴，一同度过了欢乐的三年时间。我们交流科研的灵感思路，探讨问题，他们在论文写作和学术研究方面给了我很多帮助和建议，同时感谢我的论文合作者们、师兄师弟们，祝在科研路上一帆风顺，成果多多。此外，感谢我的两位室友三年的陪伴以及对我的包容，祝前程似锦。

硕士三年，起笔是你。感谢我的女朋友陈萱一路的理解包容与陪伴。

最后，感恩我的父母和家人们，感谢他们一直以来的支持和鼓励。

二十载求学路并不是终点，路漫漫其修远兮，吾将上下而求索。

个人简历

陆承泽，出生于 1997 年 9 月 19 日。在 2020 年毕业于西安电子科技大学软件工程专业并获得学士学位。于 2020 年至今在南开大学就读计算机科学与技术硕士研究生。

研究生期间获得的学术成果:

- Li, Zhen*, **Cheng-Ze Lu***, et al. Towards an end-to-end framework for flow-guided video inpainting[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 17562-17571.
- Guo, Meng-Hao, **Cheng-Ze Lu**, et al. SegNeXt: Rethinking Convolutional Attention Design for Semantic Segmentation[C]//Advances in Neural Information Processing Systems. 2022: 1140-1156.
- Gao, Shang-Hua, Yong-Qiang Tan, Ming-Ming Cheng, **Cheng-Ze Lu**, et al. Highly efficient salient object detection with 100k parameters[C]//Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI. Cham: Springer International Publishing, 2020: 702-721.
- 刘云, **陆承泽**等. 基于高效的多尺度特征提取的轻量级语义分割 [J]. 计算机学报,2022,45(07):1517-1528.
- Hou, Qibin, **Cheng-Ze Lu**, et al. Conv2Former: A Simple Transformer-Style ConvNet for Visual Recognition[J]. arXiv preprint arXiv:2211.11943, 2022.
- Guo, Meng-Hao, **Cheng-Ze Lu**, et al. Visual attention network[J]. arXiv preprint arXiv:2202.09741, 2022.
- Huang, Zhicheng, Xiaojie Jin, **Cheng-Ze Lu**, et al. Contrastive masked autoencoders are stronger vision learners[J]. arXiv preprint arXiv:2207.13532, 2022.
- **Lu, Cheng-Ze**, et al. CMAE-V: Contrastive Masked Autoencoders for Video Action Recognition[J]. arXiv preprint arXiv:2301.06018, 2023.