

可逆式柱状网络

蔡雨萱¹Yizhuang Zhou¹韩琦¹孙建设¹孔祥文¹李俊¹Xiangyu Zhang^{12*}MEGVII 技术¹北京人工智能学会²

{caiyuxuan, zhoyizhuang, hanqi, zhangxiangyu}@megvii.com

摘要

我们提出了一种新的神经网络设计范式 *可逆列网络* (RevCol)。RevCol 的主体由多个子网络（分别命名为列）组成，子网络之间采用多级可逆连接。这种架构方案使 RevCol 的行为与传统网络大相径庭：在前向传播过程中，RevCol 中的特征在经过每一列时都会被逐渐分解，其 *tail* 信息被保留下来，而不是像其他网络那样被压缩或丢弃。我们的实验表明，CNN 风格的 RevCol 模型可以在图像分类、物体检测和语义分割等多个计算机视觉任务中取得极具竞争力的性能，尤其是在参数预算较大和数据集较大的情况下。例如，经过 ImageNet-22K 预训练后，RevCol-XL 获得了 88.2% 的 ImageNet-1K 准确率。在预训练数据较多的情况下，我们最大的模型 RevCol-H 在 ImageNet-1K 上的准确率达到 **90.0%**，在 COCO 检测最小集上的 AP_{box} 准确率达到 **63.8%**，在 ADE20k 分割上的 mIoU 准确率达到 **61.0%**。据我们所知，这是 *纯粹*（静态）CNN 模型中 COCO 检测和 ADE20k 分割结果最好的。此外，作为一种通用的宏架构时尚，RevCol 还可以被引入变压器或其他神经网络，这在计算机视觉和 NLP 任务中都能提高性能。我们在 <https://github.com/megvii-research/RevCol> 上发布了代码和模型。

1 引言

信息瓶颈原理 (IB) (Tishby 等人, 2000 年; Tishby & Zaslavsky, 2015 年) 统治着深度学习领域。请看图 1 (a) 所示的典型监督学习网络：靠近输入的层包含更多低级信息，而靠近输出的特征则具有丰富的语义。换句话说，在逐层传播的过程中，与目标无关的信息会逐渐被压缩。虽然这种学习范式在许多实际应用中取得了巨大成功，但从 *特征学习* 的角度来看，它可能并不是最佳选择--如果学习到的特征被过度压缩，或者学习到的语义信息与目标任务无关，特别是当源任务和目标任务之间存在明显的领域差距时，下游任务可能会受到影响而表现不佳 (Zamir 等人, 2018 年)。研究人员为使所学特征更具普遍适用性付出了巨大努力，例如通过自监督预训练 (Oord 等人, 2018 年; Devlin 等人, 2018 年; He 等人,



2022 年; Xie 等人, 2022 年) 或多任务学习 (Ruder, 2017 年; Caruana, 1997 年; Sener & Koltun, 2018 年)。

在本文中, 我们主要关注另一种方法: 构建一个网络来学习 *分解表征*。与 *IB* 学习不同, 分解特征学习 (Desjardins 等人, 2012 年; Bengio 等人, 2013 年; Hinton, 2021 年) 并不打算提取最相关的 信息, 而舍弃不相关的信息; 相反, 它旨在将任务相关的概念或语义词分别嵌入到几个解耦维度中。同时, 整个特征向量大致保持了与输入相同的信息量。这与生物细胞的机制 (Hinton, 2021; Lillicrap et al., 2020) 十分相似--每个细胞共享整个基因组的相同拷贝, 但表达强度不同。因此, 在计算机视觉任务中, 学习分离的特征也是

*通讯作者。本工作得到国家重点研发计划 (编号: 2017YFA0700800) 和北京人工智能学会 (BAAI) 的支持。

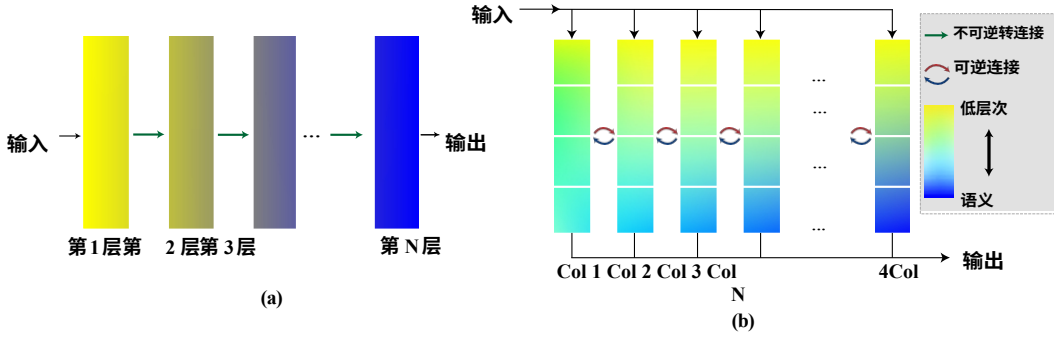


图 1：信息传播简图：(a) 香草单列网络。(b) 我们的可逆列网络。黄色表示低级信息，蓝色表示语义信息。

例如，高级语义表征在 *ImageNet* 预训练过程中进行了调整，而低级信息（如边缘位置）也应在其他特征维度中保留，以应对物体检测等下游任务的需求。

图 1 (b) 勾勒了我们的主要想法：可逆列网络 (*RevCol*) 的灵感主要来源于 *GLOM* 大图 (Hinton, 2021 年)。我们的网络由 N 个结构相同（但权重不一定相同）的子网络（名为列）组成，每个子网络接收一份输入并生成预测。因此，每一列中都存储了多级嵌入，即从低级到高级语义表征。此外，还引入了可逆变换，以便将多层次特征从第 i 列传播到第 $(i + 1)$ -列，而不会造成信息损失。在传播过程中，由于复杂性和非线性的增加，所有特征层的质量有望逐渐提高。因此，最后一列（图 1 (b) 中的第 N 列）预测的是输入的最终分解表示。

在 *RevCol* 中，我们的主要贡献之一是设计相邻列之间的可逆变换。这一概念借鉴了可逆网络家族 (Chang 等人, 2018; Gomez 等人, 2017; Jacobsen 等人, 2018; Mangalam 等人, 2022)；然而，传统的可逆结构，如 *RevNets* (Gomez et al., 2017)（图 2 (a)）通常有两个缺点：首先，可逆块内的特征图被限制为具有相同的形状*；其次，由于可逆性，*RevNets* 中的最后两个特征图必须同时包含低级和高级信息，这可能与 *IB* 原则相冲突，难以优化。本文通过引入一种新型的可逆多级融合模块来克服上述缺点。具体细节将在第 2 节中讨论。

我们在不同的复杂度预算下建立了一系列基于 CNN 的 *RevCol* 模型，并在主流计算机视觉任务中对它们进行了评估，如 *ImageNet* 分类、*COCO* 物体检测和实例分割，以及 *ADE20K* 语义分割。与复杂的 CNN 或 *ConvNeXt* (Liu 等人, 2022b) 和 *Swin* (Liu 等人, 2021) 等视觉转换器相比，我们的模型取得了相当或更好的结果。例如，经过 *ImageNet*-22K 预训练后，我们的 *RevCol*-XL 模型在 *ImageNet*-1K 上获得了 **88.2%** 的准确率，而无需使用转换器或大型卷积核 (Ding 等人, 2022b; Liu 等人, 2022b; Han 等人, 2021)。更重要的是，我们发现 *RevCol* 可以很好地扩展到大型模型和大型数据集。在一个更大的私有预训练数据集上，我们最大的模型 *RevCol*-H 在 *ImageNet*-1K 分类上分别获得了 **90.0%** 的准确率，在 *COCO* 检测最小集上获得了 **63.8%** 的 AP_{box} ，在 *ADE20K* 分割上获得了 **61.0%** 的 $mIoU$ 。据我们所知，

它是这些任务中最好的可逆模型，也是 COCO 和 ADE20K 上最好的 *纯* CNN 模型，因为它只涉及静态核，没有动态卷积 (Dai 等人, 2017 年; Ma 等人, 2020 年)。在附录中，我们进一步证明了 RevCol 可以与变换器协同工作 (Dosovitskiy 等人, 2020; Devlin 等人, 2018)，并在计算机视觉和 NLP 任务中获得更好的结果。最后，与 RevNets (Gomez 等人, 2017 年) 类似，RevCol 也具有可逆性节省内存的优点，这对于大型模型训练尤为重要。

与之前工作的关系 尽管我们关于特征分解的最初想法来自于 *GLOM* (Hinton, 2021 年)，*RevCol* 进行了大量简化和修改。例如

*准确地说，奇数索引和偶数索引的特征图应分别大小相等。

GLOM 建议采用对比辅助损失法来避免特征崩溃。对比训练法需要额外的正负样本对，既复杂又不稳定。在 RevCol 中，列之间的可逆变换本质上提供了无损的信息传播。至于其他多尺度网格状架构，如 *HRNets* (Wang 等人, 2020)、*DEQ 模型* (Bai 等人, 2020) 和 *FPN* (Lin 等人, 2017; Tan 等人, 2020)，这些模型的设计目的是融合多尺度特征，而不是学习分离的表征；因此，一般来说，它们仍然遵循图 1 (a) 中的范式--既不采用多出入口，也不采用可逆结构。基于这些网格状网络拓扑结构，基于 NAS 的工作 (Ding 等人, 2021 年; Wu 等人, 2021 年; Liu 等人, 2019 年; Ghiasi 等人, 2019 年) 为特定数据集搜索优化的网络架构拓扑结构。然而，RevCol 架构并不局限于特定任务或数据集。由于具有可逆性，我们的方法保持了无损信息传播，不仅有利于预训练，还有利于其他下游任务。最近，*RevBiFPN* (Chiley 等人, 2022 年) 提出了 FPN 的可逆变体，并进一步应用于类似 HRNet 的架构中。虽然我们的 RevCol 与 RevBiFPN 有着类似的多尺度可逆变换思想，但我们的工作是完全独立的，它源于不同的特征分解动机，具有更简单的架构（例如没有可逆上采样塔）和更高的性能。我们将在第 3 节中对其中一些模型进行比较。

2 方法

在本节中，我们将介绍 *可逆列网络 (RevCol)* 的设计细节。图 1 (b) 展示了顶层架构。请注意，对于 RevCol 中的每一列，为了简单起见，我们直接重用了现有的结构，如 *ConvNeXt* (Liu 等人, 2022b)，因此在下面的小节中，我们主要关注如何建立列之间的可逆连接。此外，我们还在每一列的顶部引入了一个即插即用的中间监督，从而进一步提高了训练收敛性和特征质量。

2.1 多层可逆装置

在我们的网络中，*可逆变换* 在无信息损失的特征分解中发挥了关键作用，其见解来自 *可逆神经网络* (Dinh 等人, 2014 年; Chang 等人, 2018 年; Gomez 等人, 2017 年; Jacobsen 等人, 2018 年; Mangalam 等人, 2022 年)。其中，我们首先对一项代表性工作 *RevNet* (Gomez 等人, 2017 年) 进行回顾。如图 2 (a) 所示，RevNet 首先将输入 x 分成两组，即 x_0 和 x_1 。然后，对于后面的块，例如块 t ，它将两个前区块的输出 x_{t-1} 和 x_{t-2} 作为输入，并生成输出 x_t 。区块 t 的映射是 *可逆的*，即 x_{t-2} 可以由两个后区块 x_{t-1} 和 x_t 重建。形式上，正演和反演计算遵循以下公式[†]：

$$\begin{aligned} \text{正向: } x_t &= F_t(x_{t-1}) + \gamma x_{t-2} \quad \text{反向} \\ &: x_{t-2} = \gamma^{-1} [x_t - F_t(x_{t-1})], \end{aligned} \quad (1)$$

其中， F_t 表示任意非线性运算，类似于标准 *ResNets* 中的残差函数； γ 是一种简单的可逆运算（例如信道缩放），其逆运算用 γ^{-1} 表示。正如引言中提到的，上述表述对特征尺寸，即 $x_t, x_{t+2}, x_{t+4}, \dots$ 必须大小相等，这在结构上并不灵活设计。这就是为什么 RevNets (戈麦斯等人, 2017 年) 在可逆单元之间引入了一些非可逆的向下采样块，因此整个网络并非完全可逆。更重要的是，我们发现没有明确的方法可以直接利用公式 1 来桥接图 1 (b) 中的列。

为了解决这个问题，我们将公式 1 概括为以下形式：

$$\begin{aligned} \text{正向: } x_t &= F_t(x_{t-1}, x_{t-2}, \dots, x_{t-m+1}) + \gamma x_{t-m} \\ \text{反向: } x_{t-m} &= \gamma^{-1} [x_t - F_t(x_{t-1}, x_{t-2}, \dots, x_{t-m+1})] \end{aligned} \quad (2)$$

其中 m 是递推的阶数 ($m \geq 2$)。显然，扩展仍然是可逆的。然后，我们将每 m 个特征图划分为一组： (x, x_{12}, \dots, x_m) ， $(x, x_{m+1m+2}, \dots, x_{2m})$ 、给定

[†]在戈麦斯等人 (2017) 的研究中，提出的可逆方程被表述为 $y_1 = xI + F(x_2)$ 和 $y_2 = x_2 + G(y_1)$ 。而在本文中，我们将这些符号 y_2 、 y_1 、 x_2 、 x_1 、 G 、 F 分别重新表述为 x_t 、 x_{t-1} 、 x_{t-2} 、 x_{t-3} 、 F_t 、 F_{t-1} ，以便更好地说明构件 t 和 $t-1$ 之间的关系。很容易证明这两个公式是等价的。

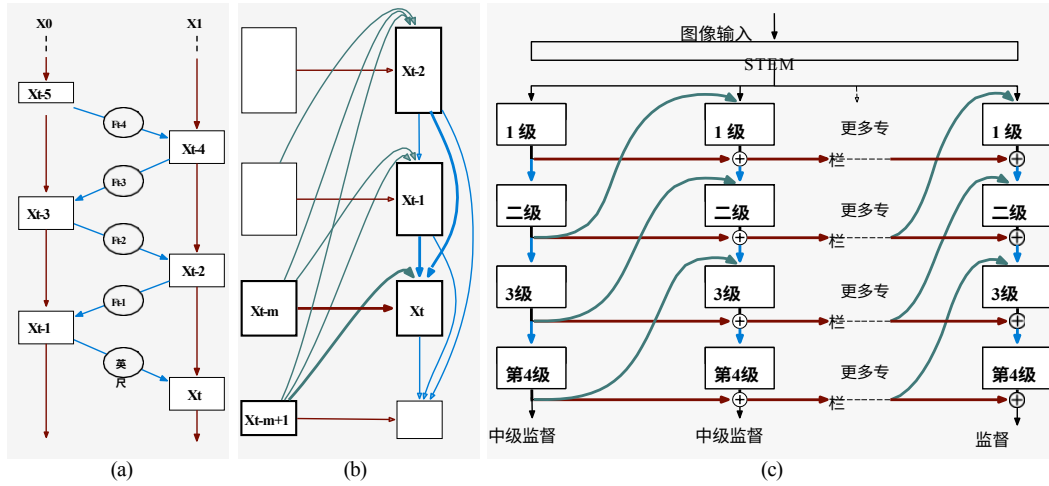


图 2: (a) RevNet 中的可逆单元 (Gomez 等人, 2017 年)。 (b) 多层可逆单元。第 t 层的所有输入均突出显示。 (c) 带有简化多级可逆单元的整个可逆列网络架构概览。

根据公式 2, 我们可以很容易地递归计算出任意一组中的特征。与原来的形式相比, 公式 2 有以下两个很好的特性:

- 如果 m 比较大, 对特征图大小的限制就会大大放宽。请注意, 公式 1 并不要求每个组内 m 特征图大小相等, 这种限制只存在于组间。因此, 我们可以使用不同形状的张量来表示不同语义层次或不同分辨率的特征。
- 公式 2 可以很容易地与现有的网络结构配合, 尽管后者是不可逆转的。例如, 我们可以在标准 *ResNet* 中分配 m 个特征图来表示一个组 $(x, x_{t+1}, \dots, x_{t+m-1})$ 中的特征图, 这仍然符合公式 2 因为 *ResNet* 可分别视为 $(F_t, F_{t+1}, \dots, F_{t+m-1})$ 的一部分。因此整个网络仍然是可逆的。

因此, 我们可以将公式 2 重组为多列式, 如图 2 (b) 所示。每一列由一组中的 m 个特征图及其母网络组成。我们将其命名为 **多级可逆单元**, 如图 1 (b) 所示, 它是我们的 *RevCol* 的基本组成部分。

2.2 可逆式立柱结构

2.2.1 宏观设计

如导言所述 (见图 1(b)), 我们的网络 *RevCol* 由多个子网络组成, 这些子网络具有 **可逆连接**, 可以进行特征分解。图 2 (c) 详细阐述了架构设计。按照近期模型的常见做法 (Dosovitskiy 等人, 2020; 刘等人, 2022b), 首先由一个补丁嵌入模块将输入图像分割成不重叠的补丁。然后, 将补丁输入每个子网络 (列)。列可以用任何传统的单列架构实现, 例如 *ViT* (Dosovitskiy 等人, 2020 年) 或 *ConvNeXt* (刘等人, 2022b)。我们从每一列中提取四级特征图, 在列之间传播信息; 例如, 如果列是用广泛使用的分层网络实现的 (刘等

人, 2021; 何等人, 2016; 刘等人, 2022b), 我们只需从每一级的输出中提取多分辨率特征即可。对于分类任务, 我们只使用最后一列中最后一级 (第 4 级) 的特征图来获取丰富的语义信息。对于物体检测和语义分割等其他下游任务, 我们使用最后一列中所有四个层级的特征图, 因为它们同时包含低层次信息和语义信息。

为了实现列之间的可逆连接, 我们采用了 *多级可逆单元* 式 2 中提出的, 但经过简化: 每个非线性参数的输入量不是 $(m - 1)$

操作 $F_t(-)$ ，我们只使用当前列的一个低级特征 x_{t-1} 和上一列的一个高级特征 x_{t-m+1} 作为输入。这种简化不会破坏可逆特性。我们发现更多的输入会带来微小的精度提升，但会消耗更多的 GPU 资源。

因此，公式 2 简化为

$$\begin{aligned} \text{正向: } x_t &= F_t(x_{t-1}, x_{t-m+1}) + \gamma x_{t-m} \\ \text{反向: } x_{t-m} &= \gamma^{-1} [x_t - F_t(x_{t-1}, x_{t-m+1})]. \end{aligned} \quad (3)$$

与传统架构相比，我们的 RevCol 的宏观设计具有以下三个特性或优势：

特征分离。在 RevCol 中，每一列的最低层保持低层特征，因为它接近输入，而最后一列的最高层则具有高度语义性，因为它直接与监督相连。因此，在列之间的（无损）传播过程中，不同层次的信息会逐渐分离--有些特征图的语义性越来越强，有些则保持低层次。详细分析见附录

E。这一特性带来了许多潜在优势，例如，对于同时依赖高层和低层特征的下游任务而言，它更具灵活性。我们认为，可逆连接在析取机制中起着关键作用--之前的一些工作，如 HRNet (Wang 等人，2020 年)，涉及多层次特征融合，但没有可逆连接，这可能会造成信息丢失，导致我们的实验表现不佳（见第 3.5.2 节）。

节省内存。由于梯度计算的需要，传统网络的训练需要占用大量内存来存储前向传播过程中的激活状态。而在我们的 RevCol 中，由于列与列之间的连接是显式可逆的，因此在反向传播过程中，我们可以即时从最后一列向第一列重建所需的激活度，这意味着在训练过程中，我们只需要在内存中保留一列的激活度。第 3.5.4 节

我们证明，随着列数的增加，RevCol 大约要多花费 $O(1)$ 的内存。

大型模型的新扩展因子。在 RevCol 架构中，除了深度（块数）和宽度（每个块的通道数）之外，列数还是虚构单列 CNN 或 ViT 的一个新维度。在一定范围内，增加列数与增加宽度和深度的效果类似。

2.2.2 微型设计

我们默认采用 ConvNeXt 块 (Liu 等人，2022b) 来实现网络中的每一列；其他架构，如转换器，也同样适用（详见附录 B）。为了使 ConvNeXt 与我们的宏架构兼容，我们做了一些修改：

融合模块。如图 5 所示，在原始 ConvNeXt 的每一级中，输入首先在一个片段合并块中进行向下采样。然后，输出经过一系列残差块。在 RevCol 中，我们引入了一个融合模块，用于融合当前列和上一列的特征图（参见图 2 (c)，绿色和蓝色连接）。我们修改了 ConvNeXt 中的原始斑块合并块，将层级规范放在斑块合并卷积之后，而不是之前。在贴片合并卷积中，通道数被加倍。我们还引入了上采样块，它由一个线性通道映射层、一个 LayerNorm 归一化层和一个特征图插值层组成。我们将线性通道映射层中的通道数减半。这两个区块的

输出相加，然后传递给残差区块。

内核大小。在 RevCol 中，我们将原 ConvNeXt (Liu 等, 2022b) 中的 7×7 卷积修改为默认的 3×3 ，主要是为了加快训练速度。增加内核大小可以进一步提高准确度，但提高幅度不大，部分原因是我们的多列设计扩大了有效感受野。

详情请参见第 3.5.5 节。

可逆操作 γ 我们采用可学习的可逆信道扩展作为可逆操作 γ ，以保持网络的稳定性。根据公式 3，每次都对特征进行求和、

的幅度越大，训练过程就越不稳定。使用可学习的缩放可以抑制特征的幅度。在训练过程中，我们会截断 γ 的绝对值，以便永远不会小于 $1e^{-3}$ ，因为当 γ 太小时，反向计算的数值误差可能会变得很大。

2.3 中间监督

虽然多级可逆单元能够在列迭代过程中保持信息，但向下采样块仍然会丢弃列内的信息。前列末端的特征与最终输出过于接近，可逆连接只需进行缩放和求和。这种信息丢失会导致性能下降。使用深度监督方法时也会出现类似问题 (Lee 等人, 2015; Szegedy 等人, 2015)。

为了缓解信息崩溃问题，我们提出了一种中间监督方法，即在前列增加额外监督。对于前列的特征，我们希望尽可能保持特征与输入图像之间的互信息，从而使网络在列内丢弃的信息更少。考虑到 RevCol 会逐渐分离语义信息和底层信息，提取和利用与任务相关的信息可以进一步提高性能。因此，我们需要最大化特征与预测之间的互信息下限。

受 Wang 等人 (2021 年) 的启发，我们为最后一级特征 (第 4 级) 添加了两个辅助头。一个是解码器 (He 等人, 2022 年)，用于重建输入图像；另一个是线性分类器。线性分类器可以通过交叉熵 (CE) 损失进行常规分类训练。解码器的参数是通过最小化二进制交叉熵 (BCE) 重建损失来优化的。与常用的 $L1$ 和 $L2$ 损失相比，将重构对数和输入图像的分布解释为比特概率 (Bernoullis) 会输出更平滑的值，这使得它更符合 CE 损失。

对于一列的中间监督，复合损失是上述两项损失的加权总和。需要注意的是，不一定所有列都有监督头。对于 RevCol 的所有变体，我们根据经验将复合损失的数量设为 4 (例如，对于 8 列 RevCol，监督头被添加到第 2、4、6 和 8 列)。

总损失 L_{in} 是所有复合损失的总和：

$$L = \sum_{i=1}^n (\alpha_i L_{iBCE} + \beta_i L_{iCE}) \quad (4)$$

n 表示复合损失总数。 L_{BCE} 和 L_{CE} 分别表示 BCE 损失和 CE 损失。 α_i 和 β_i 与复合损失数呈线性变化。当

在前几列中增加了复合损失时，我们使用较大的值 α_i 和较小的值 β_i ，以保持 α 和 β 。在后面的列中， α_i 的值减小， β_i 的值增大，这有助于提高性能。

3 实验

我们构建了不同的 RevCol 变体 (RevCol-T/S/B/L)，其复杂程度与 Swin 变压器和 ConvNeXts 相似。我们还构建了更大的 RevCol-XL 和 RevCol-H，以测试扩展能力。这些变体采用不同的通道维数 C 、每列中的块数 B 和列数 COL 。这些模型变体的配置超参数为

- RevCol-T: $C = (64, 128, 256, 512)$, $B = (2, 2, 4, 2)$, $COL = 4$
- RevCol-S: $C = (64, 128, 256, 512)$, $B = (2, 2, 4, 2)$, $COL = 8$

和历史研究国际会议 (ICLR 2023) 。

- RevCol-B: $C = (72, 144, 288, 576)$, $B = (1, 1, 3, 2)$, $COL = 16$
- RevCol-L: $C = (128, 256, 512, 1024)$, $B = (1, 2, 6, 2)$, $Col = 8$
- RevCol-XL: $C = (224, 448, 896, 1792)$, $B = (1, 2, 6, 2)$, $Col = 8$
- RevCol-H: $C = (360, 720, 1440, 2880)$, $B = (1, 2, 6, 2)$, $COL = 8$

我们在 *ImageNet* 数据集 (Deng 等人, 2009 年; Ridnik 等人, 2021 年) 上进行图像分类。

此外, 我们还在以下数据集上测试了我们的模型: 下游对象检测任务和语义分割任务。

表 1: **ImageNet 分类结果**。我们将我们的模型与具有可比 FLOP 和参数的最先进的视觉转换器和 CNN 进行了比较。↑ 表示使用大于 224² 的图像大小对模型进行微调。我们报告了 ImageNet 验证集上的前 1 位准确率以及参数和 FLOPs 的数量。我们的模型以灰色标出。

| 模型 | 图像大小 | Params (M) | FLOPs (G) | Top-1 Acc. |
|--|------------------|------------|-----------|------------|
| <i>ImageNet-1K 训练模型</i> | | | | |
| • Swin-T (Liu et al.) | 224 ² | 28 | 4.5 | 81.3 |
| • DeiT-S (Touvron et al.) | 224 ² | 22 | 4.6 | 79.8 |
| • Rev-ViT-S (Mangalam 等人) | 224 ² | 22 | 4.6 | 79.9 |
| • RevBiFPN-S3 (Chiley 等人) | 288 ² | 20 | 3.3 | 81.1 |
| • EfficientNet-B4 (Tan & Le) | 380 ² | 19 | 4.2 | 82.9 |
| • ConvNeXt-T (Liu 等人) | 224 ² | 29 | 4.5 | 82.1 |
| • RevCol-T | 224 ² | 30 | 4.5 | 82.2 |
| • Swin-S (Liu et al.) | 224 ² | 50 | 8.7 | 83.0 |
| • MViTv1-B (Fan 等人) | 224 ² | 37 | 7.8 | 83.0 |
| • T2T-ViT-19 (Yuan 等人) | 224 ² | 39 | 8.4 | 81.4 |
| • RevBiFPN-S4 (Chiley et al.) | 320 | 4910 | .6 | 83.0 |
| • EfficientNet-B5 Tan & Le) | 456 ² | 30 | 9.9 | 83.6 |
| • ConvNeXt-S (Liu 等人) | 224 ² | 50 | 8.7 | 83.1 |
| • RevCol-S | 224 ² | 60 | 9.0 | 83.5 |
| • 斯温-B (Liu 等人) | 224 ² | 89 | 15.4 | 83.5 |
| • DeiT-B (Touvron 等人) | 224 ² | 86 | 17.5 | 81.8 |
| • Rev-ViT-B (曼加拉姆 等人) | | 87 | 17.6 | 81.8 |
| • RepLKNet-31B (Ding 等人) | 224 ² | 79 | 15.3 | 83.5 |
| • RevBiFPN-S5 (Chiley 等人) | 352 ² | 82 | 21.8 | 83.7 |
| • EfficientNet-B6 (Tan & Le) | 528 ² | 43 | 19.0 | 84.0 |
| • ConvNeXt-B (Liu 等人) | 224 ² | 88 | 15.4 | 83.8 |
| • RevCol-B | 224 ² | 13816 | .6 | 84.1 |
| <i>ImageNet-22K 预训练模型 (ImageNet-1K 微调模型)</i> | | | | |
| • Swin-B (Liu et al.) | 224 ² | 8815 | .4 | 85.2 |
| • Swin-B↑ (Liu et al.) | 384 ² | | 8847.0 | 86.4 |
| • ViT-B↑ (Dosovitskiy 等人) | 384 ² | 8655 | .4 | 84.0 |
| • RepLKNet-31B (Ding et al.) | 224 ² | 7915 | .3 | 85.2 |
| • RepLKNet-31B↑ (Ding et al.) | 384 ² | | 7945.1 | 86.0 |
| • ConvNeXt-B (Liu 等人) | 224 ² | 8915 | .4 | 85.8 |
| • ConvNeXt-B↑ (Liu 等人) | 384 ² | | 8945.1 | 86.8 |
| • RevCol-B | 224 ² | 13816 | .6 | 85.6 |
| • RevCol-B↑ | 384 ² | 13848 | .9 | 86.7 |
| • Swin-L (Liu et al.) | 224 ² | | 19734.5 | 86.3 |
| • Swin-L↑ (Liu 等人) | 384 ² | 197 | 103.9 | 87.3 |
| • ViT-L↑ (Dosovitskiy 等人) | 384 ² | 307 | 190.7 | 85.2 |
| • RepLKNet-31L (Ding et al.) | 384 ² | 172 | 96.0 | 86.6 |
| • ConvNeXt-L (Liu 等人) | 224 ² | 198 | 34.4 | 86.6 |
| • ConvNeXt-L↑ (Liu 等人) | 384 ² | 198 | 101.0 | 87.5 |
| • RevCol-L | 224 ² | 27339 | .0 | 86.6 |
| • RevCol-L↑ | 384 ² | 273116.0 | 87.6 | |
| • ConvNeXt-XL↑ (Liu et al.) | 384 ² | 350 | 179.0 | 87.8 |
| • RevCol-XL↑ | 384 ² | 834 | 350.0 | 88.2 |
| <i>额外数据预训练模型 (ImageNet-1K 微调模型)</i> | | | | |
| • RevCol-XL↑ | 384 ² | | 834350.0 | |
| • RevCol-H↑ | 89.4 | | | |
| | 640 ² | 2158 | 2537 | 90.0 |

常用的 *MS-COCO* (Lin 等人, 2014 年) 和 *ADE20k* (Zhou 等人, 2017b) 数据集。训练和微调设置请参见附录 D。此外, 我们还展示了带有转换器的 RevCol 在视觉和语言任务上的表现 (见附录 B)。

3.1 图像分类

在 ImageNet (128 万张图像) (Deng 等人, 2009 年) 数据集上, 我们对 RevCol 进行了 300 次历时训练, 并有中间监督。我们还在更大的 ImageNet-22K 数据集 (Ridnik 等人, 2021 年) 上对我们的模型进行了预训练, 该数据集包含 1420 万张图像。

在表 1 中, 我们将 RevCol 变体与最近常用的变换器和 CNN 在 ImageNet-1k 验证集上进行比较。1 中, 我们将 RevCol 变体与最近在 ImageNet-1k 验证集上常用的变换器和 CNN 进行了比较。我们的模型优于大量复杂度相似的虚构单列 CNN 和变换器。例如, RevCol-S 的 Top-1 准确率为 83.5%, 比 ConvNeXt-S 高 0.4 个百分点。当使用更大的 ImageNet-22K 数据集进行预训练时, RevCol-XL 达到了 88.2% 的 Top-1 准确率。由于 RevCol 在分类预训练中保留了一些与任务无关的底层信息, 因此放宽参数和 FLOPs 的限制以及扩大数据集规模可以进一步提高模型的性能。为了进一步测试大型数据集的扩展效果, 我们建立了一个拥有

和历史研究国际会议 (ICLR 2023) 。

1.68 亿张图片的半标签数据集（见附录 C）。通过额外的数据预训练和 ImageNet-1k 微调，我们的 RevCol-H 达到了 **90.0% 的 top-1 准确率**。我们的结果进一步证明，有了 RevCol，CNN 模型也能分享大型模型和海量数据预训练的红利。

3.2 物体检测

我们在物体检测任务中评估了我们提出的 RevCol。实验是在 MS-COCO 数据集上使用级联掩码 R-CNN（Cai & Vasconcelos, 2019 年）框架进行的。我们还利用 HTC++ (Chen 等人, 2019) 和 DINO (Zhang 等人, 2022a) 框架对最大模型 RevCol-H 进行了微调。

表 2: 使用不同骨干网对 MS-COCO 数据集进行物体检测的结果。我们报告了在 COCO 最小数据集上进行单尺度测试的方框 AP 和掩码 AP。FLOPs 是在输入大小为 (1280, 800) 时测量的。

| 骨干网 | AP_{box} | AP_{box}^{50} | 美箱 75 | AP_{mask} | AP_{mask}^{50} | 美面罩 75 | 参数 | FLOPs |
|------------------------------|------------|-----------------|-------|-------------|------------------|--------|-------|-------|
| 预训练的 ImageNet-1K | | | | | | | | |
| • Swin-T (Liu 等人) | 50.5 | 69.3 | 54.9 | 43.7 | 66.6 | 47.1 | 86M | 745G |
| • ConvNeXt-T (Liu 等人) | 50.4 | 69.1 | 54.8 | 43.7 | 66.5 | 47.3 | 86M | 741G |
| • RevCol-T | 50.6 | 68.9 | 54.9 | 43.8 | 66.7 | 47.4 | 88M | 741G |
| • Swin-S (Liu 等人) | 51.8 | 70.4 | 56.3 | 44.7 | 67.9 | 48.5 | 107M | 838G |
| • ConvNeXt-S (Liu 等人) | 51.9 | 70.8 | 56.5 | 45.0 | 68.4 | 49.1 | 108M | 827G |
| • RevCol-S | 52.6 | 71.1 | 56.8 | 45.5 | 68.8 | 49.0 | 118M | 833G |
| • 斯温-B (Liu 等人) | 51.9 | 70.9 | 56.5 | 45.0 | 68.4 | 48.7 | 145M | 982G |
| • ConvNeXt-B (Liu 等人) | 52.7 | 71.3 | 57.2 | 45.6 | 68.9 | 49.5 | 146M | 964G |
| • RepLNet-B (Ding 等人) | 52.2 | - | - | 45.2 | - | - | 137M | 965G |
| • RevCol-B | 53.0 | 71.4 | 57.3 | 45.9 | 69.1 | 50.1 | 196M | 988G |
| 预训练的 ImageNet-22K | | | | | | | | |
| • 斯温-B (Liu 等人) | 53.0 | 71.8 | 57.5 | 45.8 | 69.4 | 49.7 | 145M | 982G |
| • ConvNeXt-B (Liu 等人) | 54.0 | 73.1 | 58.8 | 46.9 | 70.6 | 51.3 | 146M | 964G |
| • RepLNet-B (Ding 等人) | 53.0 | - | - | 46.3 | - | - | 137M | 965G |
| • RevCol-B | 55.0 | 73.5 | 59.7 | 47.5 | 71.1 | 51.8 | 196M | 988G |
| • Swin-L (Liu 等人) | 53.9 | 72.4 | 58.8 | 46.7 | 70.1 | 50.8 | 253M | 1382G |
| • ConvNeXt-L (Liu 等人) | 54.8 | 73.8 | 59.8 | 47.6 | 71.3 | 51.7 | 255M | 1354G |
| • RepLNet-L (Ding 等人) | 53.9 | - | - | 46.5 | - | - | 229M | 1321G |
| • RevCol-L | 55.9 | 74.1 | 60.7 | 48.4 | 71.8 | 52.8 | 330M | 1453G |
| 预先训练的额外数据 | | | | | | | | |
| • RevCol-H (HTC++) | 61.1 | 78.8 | 67.0 | 53.0 | 76.3 | 58.7 | 2.41G | 4417G |
| • RevCol-H (Objects365+DINO) | 63.8 | 81.8 | 70.2 | - | - | - | 2.18G | 4012G |

表 2 比较了 AP 和 AP 与 Swin/ConvNeXt 在 COCO 验证集上的变体大小。在表 2 中，我们比较了 AP_{box} 和 AP_{mask} 与 Swin/ConvNeXt 在 COCO 验证集上的变体大小。我们发现 RevCol 模型超越了计算复杂度类似的其他模型。预训练中保留的信息有助于 RevCol 模型在下游任务中取得更好的结果。当模型规模增大时，这一优势会变得更加显著。在 Objects365 (Shao 等人, 2019) 数据集和 DINO (Zhang 等人, 2022a) 框架下进行微调后，我们最大的模型 RevCol-H 在 COCO 检测最小集上实现了 **63.8% 的 AP_{box}** 。

表 3: 使用不同骨干网对 ADE20k 数据集进行语义分割的结果。我们报告了单/多尺度测试的 mIoU 结果。FLOP 分别是在 IN-1K 和 IN-22K 预训练模型的输入大小为 (2048, 512) 和 (2560, 640) 时测量的。

| 骨干网 | 作物面积 | $mIoU_{ss}$ | $mIoU_{ms}$ | 参数 | FLOPs |
|-----------------------|------------------|-------------|-------------|------|-------|
| ImageNet-1K 预训练 | | | | | |
| • Swin-T (Liu 等人) | 512 ² | 44.5 | 45.8 | 60M | 945G |
| • ConvNeXt-T (Liu 等人) | 512 ² | 46.0 | 46.7 | 60M | 939G |
| • RevCol-T | 512 ² | 47.4 | 47.6 | 60M | 937G |
| • Swin-S (Liu 等人) | 512 ² | 47.6 | 49.5 | 81M | 1038G |
| • ConvNeXt-S (Liu 等人) | 512 ² | 48.7 | 49.6 | 82M | 1027G |
| • RevCol-S | 512 ² | 47.9 | 49.0 | 90M | 1031G |
| • 斯温-B (Liu 等人) | 512 ² | 48.1 | 49.7 | 121M | 1188G |
| • RepLNet-B (Ding 等人) | 512 ² | 49.9 | 50.6 | 112M | 1170G |
| • ConvNeXt-B (Liu 等人) | 512 ² | 49.1 | 49.9 | 122M | 1170G |
| • RevCol-B | 512 ² | 49.0 | 50.1 | 122M | 1169G |
| 预训练的 ImageNet-22K | | | | | |
| • 斯温-B (Liu 等人) | 640 ² | 50.3 | 51.7 | 121M | 1841G |
| • RepLNet-B (Ding 等人) | 640 ² | 51.5 | 52.3 | 112M | 1829G |
| • ConvNeXt-B (Liu 等人) | 640 ² | 52.6 | 53.1 | 122M | 1828G |

| | | | | | |
|--------------------------|------------------|------|------|-------|-------|
| • RevCol-B | 640 ² | 52.7 | 53.3 | 122M | 1827G |
| • Swin-L (Liu 等人) | 640 ² | 52.1 | 53.5 | 234M | 2468G |
| • RepLKNet-L (Ding 等人) | 640 ² | 52.4 | 52.7 | 207M | 2404G |
| • ConvNeXt-L (Liu 等人) | 640 ² | 53.2 | 53.7 | 235M | 2458G |
| • RevCol-L | 640 ² | 53.4 | 53.7 | 306M | 2610G |
| <i>预先训练的额外数据</i> | | | | | |
| • RevCol-H | 640 ² | 57.8 | 58.0 | 2421M | - |
| • RevCol-H + Mask2Former | 640 ² | 60.4 | 61.0 | 2439M | - |

表 4: 采用大规模预训练的最先进视觉基础模型的系统级比较。其中包括在 *纯图像* 和 *视觉语言* 数据集上进行 *无监督* 或有 *监督* 预训练的视觉转换器、CNN 和混合架构。标有 \dagger 的 COCO 分数表示在 Object365 (Shao 等人, 2019 年) 等额外数据上进行了中间微调。

| 模型 | 参数 | 数据集 | | 图像网络 | COCO test-dev | | | ADE20K | | |
|------------|-------|-------|---------|------|---------------|----------------|----------------|-------------|------|------|
| | | 图像 | 注释 | | 检测器 | AP_{box} | AP_{mask} | 分段器 | mIoU | +ms |
| • SwinV2-G | 3.0 G | 70 M | 贴标 | 90.2 | HTC++ | 63.1 \dagger | 54.4 \dagger | UperNet | 59.3 | 59.9 |
| • BEiT3 | 1.0 G | 35 M | 标注和图像文本 | 89.6 | ViTDet | 63.7 \dagger | 54.8 \dagger | Mask2Former | 62.0 | 62.8 |
| • 佛罗伦萨 | 0.9 G | 900 M | 图像文本 | 90.1 | DyHead | 62.4 | - | - | - | - |
| • RevCol-H | 2.1 G | 168 M | 半标签 | 90.0 | DINO | 63.6 | - \dagger | Mask2Former | 60.4 | 61.0 |

3.3 语义分割

我们还利用 *UperNet* (Xiao 等人, 2018 年) 框架评估了 RevCol 主干网在 ADE20K 语义分割任务中的表现。在下流微调过程中, 我们没有使用中间监督。为了进一步探索我们模型的能力并达到领先性能, 我们使用了最近的分割框架 *Mask2Former* (Cheng 等人, 2022 年), 并采用了相同的训练设置。

在表 3 中, 我们报告了单尺度和多尺度翻转测试的验证 mIoU。在表 3 中, 我们报告了单尺度和多尺度翻转测试的验证 *mIoU*。RevCol 模型在不同的模型容量下都能获得有竞争力的性能, 这进一步验证了我们架构设计的有效性。值得注意的是, 当使用 Mask2Former 检测器和额外的预训练数据时, RevCol-H 的 mIoU 达到了 61.0%, 这显示了向大规模视觉应用扩展的可行性。

3.4 与 SOTA 基础模型的系统级比较

基础模型 (Kolesnikov 等人, 2020 年; Radford 等人, 2021 年; Yuan 等人, 2021 年b) 是在海量和多样化数据源上预先训练的通用骨干。它们可以在特定领域数据有限的情况下适应各种下游任务。我们展示了各种公开的 *最先进 (SOTA)* 基础模型 (包括视觉转换器和视觉语言模型) 之间的比较, 即 *SwinV2* (刘等人, 2022a)、*BEiT3* (王等人, 2022) 和 *Florence* (袁等人, 2021b)。如表 4 所示如表 4 所示, 尽管我们的 RevCol-H 是 *纯卷积的*, 并且在单一模态数据集上进行了预训练, 但在不同任务上的结果表明, RevCol 在大规模参数下具有显著的泛化能力。

3.5 更多分析实验

3.5.1 可逆列结构的性能增益

在本节中, 我们将评估使用可逆列所带来的性能提升。在第一个实验中, 我们先固定单列的结构和 FLOP, 然后简单地添加更多列来扩大规模并测试性能。同时, 我们还绘制了模型大小相似的虚构单列模型。如图 3 所示, 与单列模型相比, 在相同的 FLOPs 限制下, 使用多列可逆架构总能获得更好的性能。此外, 在一定范围内, 与单列模型中的块数 (深度) 和通道数 (宽度) 扩展相比, 以增加列数的方式扩展 RevCol 可以获得相似的收益。在第二

个实验中，我们将模型规模限制在约 4.5G FLOPs，并测试了不同列数的模型变体。换句话说，我们逐步增加列数，同时缩小单列规模。结果如表 5 所示。结果如表 5 所示。我们注意到，采用列数在 4 到 12 之间的模型可以保持性能，而列数进一步增加的模型则会出现性能下降。我们认为原因是单列的宽度和深度太小，无法保持表示能力。

3.5.2 可逆网络与非可逆网络非可逆网络

在本节中，我们将介绍可逆连接的不同设计模式。首先，我们利用 HRNet 的融合模块构建一个非可逆多列网络。其次，我们利用 RevNet 的设计构建了另一个单列可逆 ConvNeXt，如图 2(a)所示。我们比较了这两个

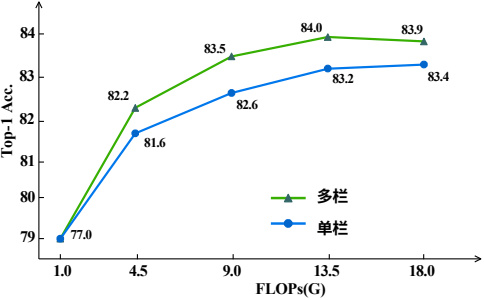


图 3: ImageNet-1K 保持单列 FLOPs 不变和增加更多列的性能。

表 5: 在计算预算相似的情况下, RevCols 中不同列数的 ImageNet 1K 性能。

| # 栏 | 参数 | FLOPs | FLOPs 每 col. | Top-1 Acc. |
|-----|-----|-------|--------------|------------|
| 1 | 28M | 4.4G | 4.40G | 81.9 |
| 4 | 30M | 4.5G | 1.12G | 82.2 |
| 8 | 34M | 4.7G | 0.59G | 82.3 |
| 12 | 33M | 4.4G | 0.35G | 82.2 |
| 20 | 35M | 4.2G | 0.21G | 81.0 |

设计。评估结果见表 6。6.不可逆的多列网络在传播过程中会出现信息丢失,从而导致精度降低。可逆单列网络在传播过程中保持了信息,但缺乏多级融合的优势。该实验进一步说明了将可逆设计与多列网络相结合的有效性。

表 6: 不同设计模式在 ImageNet-1K 上的性能比较。第 1 行表示不含可逆连接的 HRNet 风格网络。第 2 行表示 RevNet 风格网络 (不含多列)。第 3 行是我们建议的 RevCols。

| Rev. | CONN. | Params | FLOPs | Acc. |
|------|-------|--------|-------|------|
| | ✓ | 35M4 | .9G | 78.8 |
| ✓ | | 34M4 | .5G | 81.6 |
| ✓ | ✓ | 30M4 | .5G | 82.2 |

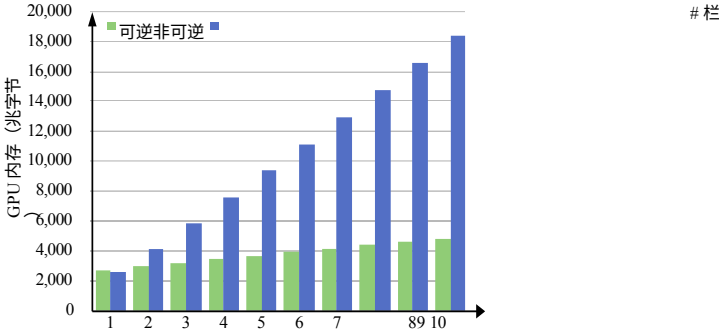
表 7: 有中间监督和无中间监督模型的性能比较。结果在 ImageNet-1K 和 COCO 数据集上报告。我们在 COCO 检测任务中使用了 1× 训练计划。

| 模型 | 中间监督 | Top-1 Acc. | APbox | APmask |
|----------|------|-------------|-------------|-------------|
| RevCol-T | C | 81.4 | 48.3 | 41.8 |
| RevCol-T | ✓ | 82.2 (+0.8) | 48.8 (+0.6) | 42.2 (+0.4) |
| RevCol-S | C | 83.0 | 50.7 | 43.8 |
| RevCol-S | ✓ | 83.5 (+0.5) | 51.1 (+0.4) | 43.8 (+0.0) |
| RevCol-B | C | 83.2 | 51.2 | 44.2 |
| RevCol-B | ✓ | 84.1 (+0.9) | 51.6 (+0.4) | 44.2 (+0.0) |

3.5.3 使用中间监督的绩效收益

在本节中,我们将评估 RevCol-T/S/B 在 ImageNet-1K 上有无中间超级视觉的性能。我们还评估了在 MS-COCO 数据集上使用 1 倍训练计划的物体检测任务性能。其他设置保持不变。验证结果见

表 7 显示,使用中间监督训练的模型的 top-1 准确率提高了 0.5%至 0.9%。从表 7 中可以看出,使用中间监督训练的模型的最高准确率提高了 0.5%到 0.9%。此外,中间监督还有利于下游任务,这进一步证明了中间监督的有效性。



作为会议论文发表于 2023 年国际比较文学
和历史研究国际会议 (ICLR 2023) 。

表 8: 采用较大核卷积

图 4: GPU 内存消耗与模型大小的关系

| 内核尺 寸 | FLOPs | Top-1 Acc | <i>AP</i> _{bbox} 1× | <i>AP</i> _{mask} 1× |
|----------|-------|--------------|---------------------------------|---------------------------------|
| 3 | 4.5G | 82.2 | 48.8 | 42.2 |
| 5 | 4.5G | 82.5 | 49.5 | 42.6 |
| 6 | 4.6G | 82.5 | 49.3 | 42.4 |
| 11 | 4.6G | 82.5 | 49.9 | 42.7 |

的模型的性能。

3.5.4 GPU 内存消耗与模型大小

图 4 显示了 GPU 内存消耗与模型大小的比例关系。我们将单列的计算复杂度固定为 1G FLOPs，然后增加列数。同时，我们测量了包括前向传播和后向传播在内的训练过程的内存消耗。我们在 Nvidia Tesla V100 GPU 上进行了实验，实验条件为批量大小 64、FP16 精度和 PyTorch 实现。随着列数的增加，我们可以看到 RevCol 保持了 $O(1)$ 的 GPU 内存消耗，而非可逆架构的内存消耗则随着列数的增加而线性增加。需要注意的是，我们的 RevCol 并没有严格保持相同的 GPU 内存消耗量。

随着列数的增加，可逆网络在计算梯度和反向传播中重新构建特征图时，需要备份运算权重，因此会产生消耗。

3.5.5 在卷积中消融内核大小

在最初的 ConvNeXt 中，大核卷积取得了更好的性能。我们在 RevCol-T 中进行了实验。如表 8 所示如表 8 所示，对于 4 列模型，使用 5×5 卷积可使 RevCol-T 模型的 ImageNet-1k Top-1 准确率提高 0.3%，COCO AP_{box} 提高 0.7%。进一步

增加内核大小可以提高下游任务的准确度，但不会太高。我们认为，RevCol 的设计已经扩大了有效感受野，这限制了 RevCol 的精度增益。

使用大核卷积。另一方面， 3×3 卷积在（预）训练中具有高效和稳定的优点。因此，我们在所有 RevCol 模型中都采用了核 3。

4 相关作品

4.1 将表征学习与部分-整体层次结构分开

分离表征一般被描述为分离变异因素，明确表示数据的重要属性 (Desjardins 等人, 2012 年; Bengio 等人, 2013 年)。Desjardins 等人 (2012) ; Kulkarni 等人 (2015) ; Higgins 等人 (2017) ; Chen 等人 (2016) ; Karras 等人 (2019) 试图通过生成模型来学习分离表征。Locatello 等人 (2019) 指出，如果所考虑的学习方法和数据集没有归纳偏差，无监督地学习不相干表征从根本上是不可能的。最近提出的 *GLOM* (Hinton, 2021 年) 提供了一种通过权重共享列来表示部分-整体层次结构的方法。GLOM 架构为深度神经网络提供了可解释的整体部分层次结构 (Garau 等人, 2022 年)。在 RevCol 中，我们采用了使用列的设计，但没有对岛屿的形成过程进行建模。相反，我们的列迭代过程既保留了低层信息，也保留了高层信息，并逐渐将它们分离开来。与使用自监督方法相比，RevCol 可以在端到端的监督下进行训练。

4.2 可逆网络

Gomez 等人 (2017 年) 首次提出了允许反向传播而不保存中间激活的 *RevNet*。这种可逆设计大大节省了训练成本，因为随着模型深度的增加，它能保持 $O(1)$ 的 GPU 内存消耗。Jacobsen 等人 (2018 年) 提出了一种完全可逆的网络，它可以

可以反向回到输入，而不会有任何信息丢失。Chang 等人 (2018) 建立了一个关于深度神经网络稳定性和可逆性的理论框架，并推导出可以任意深入的可逆网络。Mangalam 等人 (

2022 年) 将可逆网络的范围从 CNN 扩展到 Transformers。 *RevBiFPN* (Chiley 等人, 2022 年) 是我们的同期研究成果, 它在 *BiFPN* (Tan 等人, 2020 年) 网络中加入了可逆连接。在 *RevBiFPN* 中, 我们的 *RevCol* 维护的是每一列内的无损信息, 而不是整个 *BiFPN* 网络。

5 结 论

在本文中, 我们提出了基于可逆列的基础模型设计范式 *RevCol*。在通过列进行无损传播的过程中, *RevCol* 中的特征会被逐渐分解, 总信息量仍会保持不变, 而不会被压缩。我们的实验表明, *RevCol* 可以在多个计算机视觉任务中实现具有竞争力的性能。我们希望 *RevCol* 能在视觉和语言领域的各种任务中为提高性能做出贡献。

参考资料

白少杰、Vladlen Koltun 和 J Zico Kolter。多尺度深度平衡模型。《神经信息处理系统进展》，33:5238-5250, 2020 年。

Hangbo Bao, Li Dong, and Furu Wei. Beit: 图像变换器的伯特预训练。 *ArXiv 预印本 arXiv:2106.08254*, 2021.

Yoshua Bengio、Aaron Courville 和 Pascal Vincent。表征学习：回顾与新视角。 *IEEE patterns analysis and machine intelligence*, 35(8):1798-1828, 2013.

Zhaowei Cai 和 Nuno Vasconcelos. 级联 R-CNN: 高质量对象检测和实例分割。 *IEEE patterns analysis and machine intelligence*, 43(5):1483-1498, 2019.

里奇-卡鲁阿纳多任务学习。 *机器学习*, 28 (1) : 41-75, 1997.

Bo Chang, Lili Meng, Eldad Haber, Lars Ruthotto, David Begert, and Elliot Holtham. 任意深度残差神经网络的可逆架构。 *美国人工智能学会会议论文集*, 第 32 卷, 2018 年。

陈凯、庞江苗、王佳琪、熊宇、李潇潇、孙树阳、冯万森、刘紫薇、史建平、欧阳万里等：实例分割的混合任务级联。 In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.

Tianqi Chen、Ian Goodfellow 和 Jonathon Shlens。Net2net: *ArXiv preprint arXiv:1511.05641*, 2015.

Xi Chen、Yan Duan、Rein Houthoofd、John Schulman、Ilya Sutskever 和 Pieter Abbeel。Info-gan: 信息最大化生成对抗网的可解释表征学习。 *神经信息处理系统进展》*, 2016 年第 29 期。

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. 用于通用图像分割的掩码-注意力掩码转换器。 *IEEE/CVF 计算机视觉与模式识别会议论文集》*, 第 1290-1299 页, 2022 年。

Vitaliy Chiley、Vithursan Thangarasa、Abhay Gupta、Anshul Samar、Joel Hestness 和 Dennis DeCoste。RevBifpn: *ArXiv preprint arXiv:2206.14098*, 2022.

戴继锋、齐浩志、熊宇文、李毅、张国栋、胡涵、魏一晨可变形卷积网络 *电气和电子工程师学会计算机视觉国际会议论文集》*、pp.764-773, 2017.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: 大规模分层图像数据库。 *2009 年 IEEE 计算机视觉与模式识别会议*、pp.248-255. IEEE, 2009.

Guillaume Desjardins, Aaron Courville, and Yoshua Bengio. 通过生成纠缠解散变异因子。 *arXiv preprint arXiv:1210.5474*, 2012.

Jacob Devlin, Ming-Wei Chang, Kenton Lee 和 Kristina Toutanova. 伯特：用于语言理解的深度双向变换器预训练》, *arXiv preprint arXiv:1810.04805*, 2018.

丁明宇、连晓晨、杨林杰、王鹏、金晓杰、吕志武、罗平。Hr-nas：利用轻量级变压器搜索高效高分辨率神经架构。《IEEE/CVF 计算机视觉与模式识别会议论文集》，第 2982-2992 页，2021 年。

Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. 将内核扩大到 31x31：重新审视 cnns 中的大内核设计。《IEEE/CVF 计算机视觉与模式识别会议论文集》，第 11963-11975 页，2022a。

Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. 将内核扩大到 31x31: 重新审视 cnns 中的大内核设计。 *IEEE/CVF 计算机视觉与模式识别会议论文集*，第 11963-11975 页，2022b。

Laurent Dinh、David Krueger 和 Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Alexey Dosovitskiy、Lucas Beyer、Alexander Kolesnikov、Dirk Weissenborn、Xiaohua Zhai、Thomas Unterthiner、Mostafa Dehghani、Matthias Minderer、Georg Heigold、Sylvain Gelly 等。图像胜过 16x16 个单词: *ArXiv preprint arXiv:2010.11929*, 2020.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. 多尺度视觉转换器。 *IEEE/CVF 计算机视觉国际会议论文集*，第 6824-6835 页，2021 年。

尼古拉-加劳、尼古拉-比萨尼奥、泽诺-桑布加罗和尼古拉-孔奇。神经网络中可解释的部分-整体层次和概念-语义关系。 *IEEE/CVF 计算机视觉与模式识别会议论文集*，第 13689-13698 页，2022 年。

Golnaz Ghiasi、Tsung-Yi Lin 和 Quoc V Le. Nas-fpn: 学习用于物体检测的可扩展特征金字塔结构。 In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp.

Golnaz Ghiasi、Barret Zoph、Ekin D Cubuk、Quoc V Le 和 Tsung-Yi Lin. 学习一般表征的多任务自我训练。 *IEEE/CVF 计算机视觉国际会议论文集*，第 8856-8865 页，2021 年。

艾丹-N-戈麦斯、任梦晔、拉奎尔-乌塔松和罗杰-B-格罗斯。可逆残差网络: 不存储激活的反向传播。 *神经信息处理系统进展*，2017年第30期。

Qi Han, Zejia Fan, Qi Dai, Lei Sun, Ming-Ming Cheng, Jiaying Liu, and Jingdong Wang. 论局部注意力与动态深度卷积之间的联系 *学习表征国际会议*，2021年。

何开明、张翔宇、任少清和孙健。图像识别的深度残差学习 *电气和电子工程师学会计算机视觉与模式识别会议论文集*、pp.770-778, 2016.

何开明、陈新磊、谢赛宁、李阳浩、Piotr Dollár 和 Ross Girshick. 遮蔽式自动编码器是可扩展的视觉学习器。 *IEEE/CVF 计算机视觉与模式识别会议论文集*，第 16000-16009 页，2022 年。

Irina Higgins、Loic Matthey、Arka Pal、Christopher Burgess、Xavier Glorot、Matthew Botvinick、Shakir Mohamed 和 Alexander Lerchner. β -VAE: 用受限变异框架学习基本视觉概念。2017年 *学习表征国际会议*。URL <https://openreview.net/forum?id=Sy2fzU9gl>.

Geoffrey Hinton. 如何在神经网络中表示部分 - 整体层次结构》, *arXiv preprint arXiv:2102.12627*, 2021.

Jörn-Henrik Jacobsen、Arnold Smeulders 和 Edouard Oyallon. i-revnet: 深度可逆网络。
arXiv preprint arXiv:1802.07088, 2018.

姜鹏涛、张昌斌、侯启斌、程明明和魏云超。Layercam: 用于定位的分层类激活图。 *IEEE 图像处理论文集*, 30:5875-5888, 2021.

Tero Karras、Samuli Laine 和 Timo Aila. 基于风格的生成式对抗网络生成器架构。 In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp.

Alexander Kolesnikov、Lucas Beyer、Xiaohua Zhai、Joan Puigcerver、Jessica Yung、Sylvain Gelly 和 Neil Houlsby。大转移（比特）：通用视觉表征学习。《欧洲计算机视觉会议》，第 491-507 页。Springer, 2020.

西蒙-科恩布利斯、穆罕默德-诺鲁兹、李鸿乐和杰弗里-辛顿。再论神经网络表征的相似性。《国际机器学习会议》，第 3519-3529 页。PMLR, 2019。

Tejas D Kulkarni、William F Whitney、Pushmeet Kohli 和 Josh Tenenbaum。深度卷积反演图形网络。《神经信息处理系统进展》，2015年第28期。

Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu.深度监督网络。《人工智能与统计学》，第 562-570 页。PMLR, 2015。

Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton.逆向思维与大脑。《自然评论神经科学》，21（6）：335-346，2020 年。

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick.微软 coco：上下文中的常见对象。《欧洲计算机视觉会议》，第 740-755 页。Springer, 2014.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie.用于物体检测的特征金字塔网络。In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.

Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan L Yuille, and Li Fei-Fei.Auto-deeplab：用于语义图像分割的分层神经架构搜索。In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp.

Ze Liu、Yutong Lin、Yue Cao、Han Hu、Yixuan Wei、Zheng Zhang、Stephen Lin 和 Baining Guo。Swin 变换器：使用移位窗口的分层视觉变换器。In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2：扩大容量和分辨率。In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.

刘壮、毛汉子、吴超元、克里斯托夫-费希滕霍夫、特雷弗-达雷尔和谢赛宁。面向 2020 年代的 convnet。《IEEE/CVF 计算机视觉与模式识别会议论文集》，第 11976-11986 页，2022b。

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem.挑战无监督表征学习中的常见假设。《国际机器学习会议》，第 4114-4124 页。PMLR, 2019。

Ningning Ma, Xiangyu Zhang, Jiawei Huang, and Jian Sun.权重网络：重新审视权重网络的设

计空间。《欧洲计算机视觉会议》，第 776-792 页。Springer, 2020.

Dhruv Mahajan、Ross Girshick、Vignesh Ramanathan、Kaiming He、Manohar Paluri、Yixuan Li、Ashwin Bharambe 和 Laurens Van Der Maaten。探索弱监督预训练的极限。《欧洲计算机视觉会议 (ECCV) 论文集》，第 181-196 页，2018 年。

Karttikeya Mangalam、范昊琦、李阳浩、吴超远、熊波、Christoph Feichtenhofer 和 Jitendra Malik。可逆视觉变换器。《IEEE/CVF 计算机视觉与模式识别会议论文集》，第 10830-10840 页，2022 年。

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 具有对比预测编码的表征学习》, *arXiv preprint arXiv:1807.03748*, 2018.

Myle Ott、Sergey Edunov、David Grangier 和 Michael Auli。扩展神经机器翻译。《第三届机器翻译大会论文集：研究论文集》，第 1-9 页，比利时布鲁塞尔，2018 年 10 月。计算语言学协会。doi: 10.18653/v1/W18-6301。

Alec Radford、Jong Wook Kim、Chris Hallacy、Aditya Ramesh、Gabriel Goh、Sandhini Agarwal、Girish Sastry、Amanda Askell、Pamela Mishkin、Jack Clark 等。从自然语言监督中学习可转移的视觉模型。《国际机器学习大会》，第 8748-8763 页。PMLR，2021 年。

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. 面向大众的 Imagenet-21k 预训练。《ArXiv 预印本 arXiv:2104.10972》，2021。

Sebastian Ruder. 深度神经网络中的多任务学习概述。《arXiv preprint arXiv:1706.05098》，2017。

Ozan Sener 和 Vladlen Koltun. 作为多目标优化的多任务学习。《神经信息处理系统进展》，2018 年第 31 期。

Rico Sennrich、Barry Haddow 和 Alexandra Birch. 使用子词单元的罕见词神经机器翻译。《第 54 届计算语言学协会年会》，第 1715-1725 页。计算语言学协会 (ACL)，2016 年。

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: 用于物体检测的大规模高质量数据集。In *Proceedings of the IEEE/CVF international conference on computer vision*, pp.

克里斯蒂安-塞格迪 (Christian Szegedy)、刘伟、贾洋清、皮埃尔-塞曼内 (Pierre Sermanet)、斯科特-里德 (Scott Reed)、德拉戈米尔-安格洛夫 (Dragomir Anguelov)、杜米特鲁-埃尔汗 (Dumitru Erhan)、文森特-万胡克 (Vincent Vanhoucke) 和安德鲁-拉比诺维奇 (Andrew Rabinovich)。深入卷积。In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.

Mingxing Tan 和 Quoc Le. Efficientnet: 反思卷积神经网络的模型缩放。《国际机器学习会议》，第 6105-6114 页。PMLR，2019。

Mingxing Tan、Ruoming Pang 和 Quoc V Le. Efficientdet: 可扩展的高效物体检测。《IEEE/CVF 计算机视觉与模式识别会议论文集》，第 10781-10790 页，2020 年。

巴特-托米、戴维-A-沙马、杰拉尔德-弗里德兰、本杰明-埃利萨尔德、卡尔-倪、道格拉斯-波兰、达米安-博思、李力嘉。Yfcc100m: 多媒体研究的新数据。《ACM 通信》，59 (2): 64-73，2016 年。

纳夫塔利-提什比和诺加-扎斯拉夫斯基。深度学习与信息瓶颈原理。In *2015 IEEE information theory workshop (itw)*, pp. IEEE, 2015。

Naftali Tishby, Fernando C Pereira, and William Bialek. 信息瓶颈法。《ArXiv preprint

和历史研究国际会议 (ICLR 2023) 。

physics/0004057, 2000.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 通过注意力训练数据高效图像变换器和蒸馏器》。 *arXiv 预印本 arXiv:2012.12877*, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser 和 Illia Polosukhin. 注意力就是你所需要的一切。 *神经信息处理系统进展* 》， 2017年30期。

王敬东、孙珂、程天恒、蒋博瑞、邓超瑞、赵阳、刘东、穆亚东、谭明奎、王兴刚等：视觉识别的深度高分辨率表示学习。 *IEEE patterns analysis and machine intelligence*, 43(10):3349-3364, 2020.

王文辉、包杭波、董丽、Johan Bjorck、彭志良、刘强、Kriti Aggarwal、Owais Khan Mohammed、Saksham Singhal、Subhojit Som 等。图像作为外语：针对所有视觉和视觉语言任务的 Beit 预训练。 *ArXiv 预印本 arXiv:2208.10442*, 2022.

王玉林、倪赞林、宋世基、杨乐、黄高。重新审视局部监督学习：端到端训练的替代方案》, *arXiv preprint arXiv:2101.10832*, 2021.

吴碧晨、李朝建、张航、戴晓亮、张培朝、余马修、王嘉良、林颖妍和彼得-瓦伊达。Fbnetv5：一次运行多个任务的神经架构搜索。 *arXiv preprint arXiv:2111.10007*, 2021.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. 场景理解的统一感知解析。 *欧洲计算机视觉会议 (ECCV) 论文集*，第 418-434 页，2018 年。

Zhenda Xie、Zheng Zhang、Yue Cao、Yutong Lin、Jianmin Bao、Zhuliang Yao、Qi Dai 和 Han Hu。Simmim：遮蔽图像建模的简单框架。 In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.

I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. 用于图像分类的 Billion-scale 半监督学习。 *ArXiv preprint arXiv:1905.00546*, 2019.

Li Yuan、Yunpeng Chen、Tao Wang、Weihao Yu、YuJun Shi、Zi-Hang Jiang、Francis EH Tay、Jiashi Feng 和 Shuicheng Yan。令牌到令牌的维度：在图像网络上从头开始训练视觉转换器。 In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence： *arXiv preprint arXiv:2111.11432*, 2021b.

Amir R Zamir、Alexander Sax、William Shen、Leonidas J Guibas、Jitendra Malik 和 Silvio Savarese。任务分类学：分解任务迁移学习。 In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino：用于端到端物体检测的带改进去噪锚框的 Detr。 *arXiv 预印本 arXiv:2203.03605*, 2022a.

张元汉、孙庆红、周益春、何泽新、尹振飞、王坤、盛璐、乔宇、邵晶、刘紫薇。竹子：利用人机协同持续构建超大规模视觉数据集，2022b.

Bolei Zhou、Agata Lapedriza、Aditya Khosla、Aude Oliva 和 Antonio Torralba。地点用于场景识别的千万级图像数据库。 *IEEE Transactions on Pattern Analysis and Machine Intelligence*，2017a.

周博磊、赵航、泽维尔-普伊格、萨尼亚-菲德勒、阿德拉-巴里乌索和安东尼奥-托拉尔巴。通过 ade20k 数据集进行场景解析。 In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.

A 微型设计细节

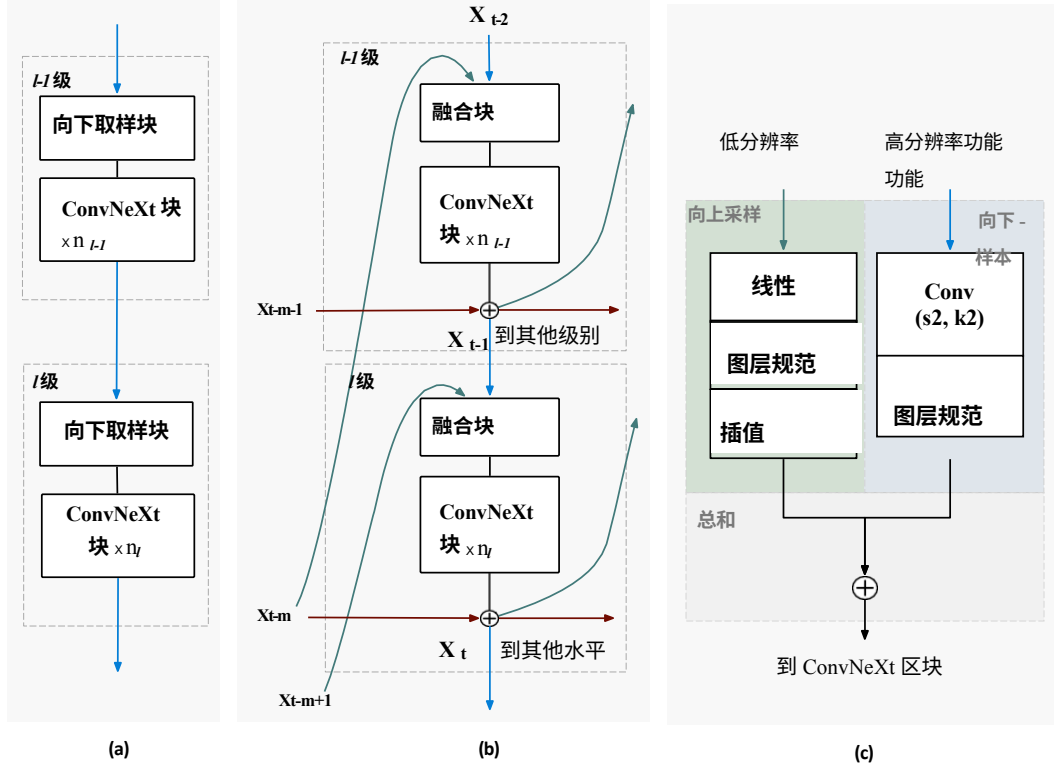


图 5: (a) ConvNeXt 中的层级。第 l 层包含一个补丁合并向下采样块和 n_l 个残差块。
(b) RevCol 中的层级。层级 l 由一个融合模块、 n_l 个残差块和一个可逆操作组成。请注意，第 l 层将特征图 x_{t-1} 、 x_{t-m+1} 和 x_{t-m} 作为输入。
特征图 x_{t-1} 和 x_{t-m+1} 输入融合模块，特征图 x_{t-m} 输入融合模块。
可逆操作。(c) 融合模块的设计。

如图 2 和第 2.2 节所示，我们的 RevCol 包含多列可逆连接。图 5 (a) 显示了 ConvNeXt 的架构。请注意，我们将 ConvNeXt 中的 7×7 深度卷积替换为

3×3 ，如第 2.2.2 节所述。在图 5 (b) 中，我们详细展示了如何将我们的 RevCol 扩展到 ConvNeXt 的基础。首先，我们将下采样块替换为融合块，以融合低级

图 5 (c) 显示了融合块的细节，其中包含上采样和下采样操作，以处理不同的分辨率。其次，对于每个层级，上一列的同层级表示都会添加到当前层级的输出中，并准备作为一个整体进行传播。得益于这两项修改，来自不同层次的特征图可以聚合在一起，形成中间表示。在图 5 (c) 中，我们使用线性层规范 (Linear-LayerNorm)，然后使用最近插值 (nearest interpolation) 对低层进行向上采样。

分辨率特征。步长为 2 的 2×2 内核 Conv2d 对高分辨率特征进行下采样、然后是 LayerNorm，以平衡两个输入的贡献。

B 适用于变压器

B.1 视觉变压器型号

RevCol 包含多个具有可逆连接的轻量级子网络。在本文中，除了 2.2.2 节所述的多列融合和更小的卷积核之外，我们默认采用 ConvNext 微设计。不过，RevCol 的微型设计并不局限于卷积网络，它也与各向同性设计兼容，例如 vanilla 视觉网络。

变压器 (ViT) (Dosovitskiy 等人, 2020 年)。在本节中, 我们展示了 RevCol 的微型设计可以推广到 vanilla ViT, 即 RevCol-ViT, 并取得了很好的实验结果。

net-ViT 保持了可逆列的特征分辨率。因此, 融合模块中的补丁合并块和上采样块都被简单的线性投影和后层规范所取代。我们使用 vanilla ViT 构建模块, 而不是 ConvNext 构建模块变体。与 Liu 等人 (2022a) 的做法类似, 在 ViT 模块中使用后 LayerNorms 和归一化点积注意来稳定训练收敛。利用各向同性的特性, 我们在每一列中均匀地排列积木块。RevCol-ViT 的配置细节如下

- RevCol-ViT-S: $C = (224, 224, 224, 224)$, $B = (2, 2, 2, 2)$, $HEAD = 4$, $COL = 4$
- RevCol-ViT-B: $C = (384, 384, 384, 384)$, $B = (3, 3, 3, 3)$, $HEAD = 6$, $COL = 4$

表 9: ImageNet-1K 分类结果。我们将 RevCol-ViT 与具有可比 FLOP 和参数的最先进的各向同性视觉变换器和 CNN 进行了比较。

| 模型 | 图像大小 | 参数 | FLOPs | Top-1 Acc. |
|--|------------------|-----|-------|-------------|
| • DeiT-S (Touvron 等人, 2020 年) | 224 ² | 22M | 4.6G | 79.8 |
| • ConvNext-S (<i>iso.</i>) (Liu et al., 2022b) | 224 ² | 22M | 4.3G | 79.7 |
| • RevCol-ViT-S | 224 ² | 16M | 4.6G | 80.6 |
| • ViT-B (Dosovitskiy 等人, 2020 年) | 384 ² | 86M | 55.4G | 77.9 |
| • DeiT-B (Touvron 等人, 2020 年) | 224 ² | 86M | 17.6G | 81.7 |
| • Rev-ViT-B (Mangalam 等人, 2022 年) | 224 ² | 87M | 17.6G | 81.8 |
| • Rev-MViT-B (Mangalam 等人, 2022 年) | 224 ² | 39M | 8.7G | 82.5 |
| • ConvNext-B (<i>iso.</i>) (Liu et al., 2022b) | 224 ² | 87M | 16.9G | 82.0 |
| • RevCol-ViT-B | 224 ² | 67M | 18.8G | 82.7 |

我们使用与第 3.1 节所述的各向异性 RevCol 相同的训练设置, 只是为了简单起见放弃了中间监督, 并将 RevCol-B 的随机深度率设置为 0.2。我们在初始化时根据网络深度缩减每个 FFN 中最后一个线性投影层的值, 这与 BEiT (Bao 等, 2021 年) 相同。在表 9 中, 我们比较了 RevCol-B、RevCol-C 和 RevCol-D。在表 9 中, 我们将 RevCol-ViT 与 vanilla ViT 和其他并发各向同性设计进行了比较。我们的 RevCol-ViT 在 ImageNet-1k 分类的 top-1 准确率上超过了具有相似模型参数和计算开销的 vanilla 视觉转换器 (ViT 为 77.9%, DeiT 为 81.7%) 和卷积网络 ConvNeXt (82.0%)。

B.2 语言模型

考虑到将变换器应用于计算机视觉 (即 ViT, Dosovitskiy 等人, 2020 年) 所取得的巨大成功, 我们也对将 RevCol 推广到自然语言处理 (NLP) 进行了一些探索。基于附录 B.1 中的设计, 我们只需稍加修改, 就能轻松地将各向同性 RevCol 应用于语言模型。具体来说, 我们将 RevCol 中的词干模块替换为转换器中的词嵌入和位置编码。然后, RevCol 就可以作为编码器插入到原来的转换器中。RevCol 最后一列的输出将用作解码器中注意层的记忆键和

值，这与原始转换器完全相同。

我们选择翻译任务来评估 RevCol 在 NLP 中的潜力。我们在包含 450 万个句子的 WMT'16 英德 (En-De) 数据集和包含 3,600 万个句子的更大 WMT'14 英法数据集上进行了实验。每个句子都按照 Sennrich 等人 (2016 年) 的方法进行了源和目标字节对联合编码。模型架构和 BLEU 得分详情见表 10。10.

所有的数据集准备和训练配置都遵循 Ott 等人 (2018 年) 和开源项目 `fairseq`。为简化起见，我们舍弃了中间监督。如表 10 所示如表 10 所示，在 En-De (28.67 对 28.43) 和 En-Fr (43.40 对 43.07) 上，我们的 RevCol 优于具有可比参数的 vanilla 变换器，这表明 RevCol 适用于 NLP。

表 10: WMT 英语-德语 (En-De) 和英语-法语 (En-Fr) 翻译任务 newstest2014 的 BLEU 分数。[†] 表示我们使用 fairseq 重新进行了实验。

| 型号 | 编码器 | | | | 解码器 | | | | 参数 | 任务 | BLEU | |
|------------|--|---|-------|------|-----|----------|-------|------|----|------|-------|--------------|
| | 拱形 | 模型 | dff | 率 | 拱门 | $dmodel$ | dff | 率领 | | | | |
| 大 | Transformer [†] (Vaswani 等人, 2017 年) | $N = 6$ $COL = 4$ $B = (1,1,1,1)$ | 1024 | 4096 | 16 | $N = 6$ | 1024 | 4096 | 16 | 209M | En-De | 28.43 |
| | | | | | | | | | | 221M | En-Fr | 43.07 |
| | | | | | | | | | | 200M | En-De | 28.67 |
| | | | | | | | | | | 209M | En-Fr | 43.40 |
| | | | | | | | | | | | | |
| RevCol-变压器 | | | 768 | 3072 | 12 | $N = 6$ | 768 | 3072 | 12 | | | |

B.3 列数的鲁棒性

在本文的消融分析中，我们发现当固定总 FLOPs 并增加 RevCol 的列数时，性能会先上升后饱和。当列数达到极大值（如 20 列）时，由于单列的表示能力有限，性能会下降。列数有限。当列数相同时，如 41²，性能相似、验证了列数设置的稳健性。

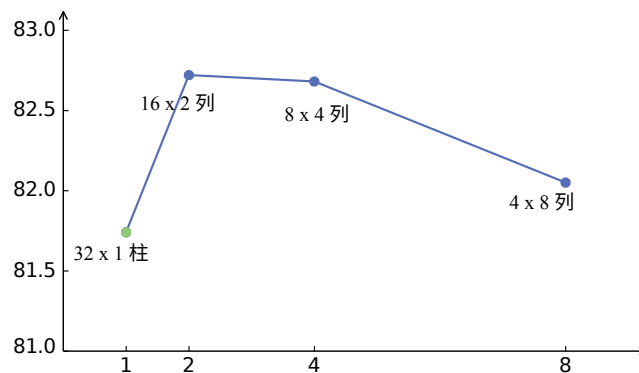


图 6: RevCol-ViT-B 不同变体的 ImageNet Top-1 准确率。每个变体的残差块总数和通道维度相同。

为了进一步分析列数的稳健性，我们在本节中建立了一些 RevCol- ViT-B 变体（详见附录 B）。每个变体的残差块数量相同，通道维度相同，但列数不同。换句话说，这些变体具有相同的通道维度、不同的每列深度和不同的列数。我们总共使用了 32 个残差块，FLOPs 保持在 18G 左右。图 6 显示了不同变体在 ImageNet-1K 上的性能。列数分别为 1、2、4 和 8，每列的深度分别为 32、16、8 和 4。单列变体的性能较低（类似于 DeiT-B (Touvron 等人, 2020 年)），因为单列 ViT 不能像多列可逆那样保持信息。当列数变多时，性能会下降，因为每一列的深度不够。这一现象表明，在给定目标 FLOPs 的情况下，除非每列深度或通道维度太小，否则列数的设置是稳健的。

C 用于大型模型的半标注私人收集数据集

C.1 数据收集和伪标签系统

该数据集包含约 1.68 亿 (M) 张图像，其中 5000 万张已标注，其余 1.18 亿张未标注。大部分标注图像来自公共数据集，如 ImageNet、Places365 (Zhou 等人, 2017a) 和 Bamboo (Zhang 等人, 2022b)。其他则是由室内员工注释的网络抓取图像。未标注的图像来自于弱注释图像文本数据集，如 YFCC-100M (Thomee 等人, 2016 年)。我们不使用文本注释。

为了利用不同标签域的图像和大量未标签图像，我们采用了与 Ding 等人 (2022a) 和 Ghiasi 等人 (2021) 类似的多目标标签系统。我们通过 ViTs 采用半监督学习策略，从而生成质量不断提高的伪标签。我们只存储置信度高于 1% 的软预测，以节省存储空间。我们使用的伪标签最终版本是由多头 ViT-Huge 教师生成的，其准确率为 89.0% ImageNet-1k。

C.2 图像重复数据删除

由于数据集包含大量未经验证的网络抓取图片，因此很可能有验证或测试图片偷偷进入我们的训练数据集。Mahajan 等人 (2018 年) 和 Yalniz 等人 (2019 年) 等著作都认为重复数据删除是公平实验的重要程序。

我们首先遍历整个数据集，根据伪标签距离过滤出可疑的重复图像以及相应的测试图像。这样就有 10,000 多张图像具有高度可疑性。我们对这些图像对进行研究，最终发现了约 1200 个完全重复和近似重复的图像。图 7 显示了一些难以检测到的近似重复的例子。尽管如此，在我们的实验中，在不去除这些重复图像的情况下训练模型，在 ImageNet-1k 上的准确率提高不到 0.1%。我们将此归因于这些重复数据中缺乏真实标签。



图 7：上图：在无标签图像中发现的近似重复图像。下图ImageNet-1k 验证图像。

D 更多培训详情

本节将介绍有关 ImageNet 分类、COCO 检测和 ADE20K 分割的更多训练细节。

D.1 中间监督设置

我们在 ImageNet-1k 训练、ImageNet-22k 和额外数据预训练中添加了中间监督。在 ImageNet-1k 训练中，我们使用了带有逐步上采样特征图的 3 块解码器。在 ImageNet-22k 和额外数据预训练中，我们使用单层解码器。对于 RevCol 的所有变体，我们根据经验将复合损失 n 的数量设为 3（例如，对于 8 列 RevCol，中间监督被添加到第 2、4 和 6 列，原始分类 CE 损失也被添加到第 8 列）。 α_i 设置为 3、2、1、0， β_i 设置为 0.18、0.35、0.53、1。

D.2 用于训练和预训练的超参数

本节将介绍主要实验的训练细节、ImageNet 上的监督训练和额外数据。我们在表 11 中显示了这一设置。11.除附加说明外，消融研究中的所有实验都在 ImageNet-1K 上进行了监督训练，并遵循本节所述设置。

表 11：训练和预训练 RevCol 的超参数。

| 超参数 | ImageNet-1K | | ImageNet- |
|------------------|------------------|--------|------------------|
| | 22K168M 额外数据 | | XL/H |
| | T/S/B | B/L/XL | |
| 输入分辨率 | 224 ² | | 224 ² |
| 训练历元 | 300 | 90 | 10 |
| 热身时间 | 20 | 5 | 0.15 |
| 批量大小 | 4096 | | 5120 |
| 峰值学习率 | 4e-3 | 5e-4 | 6.25e-4 |
| 学习率时间表 | 余弦值 | | 余弦值 |
| 分层学习率衰减 | C | | C |
| AdamW 势头 | (0.9, 0.999) | | (0.9, 0.999) |
| 重量衰减 | 0.05 | 0.1 | 0.05 |
| 渐变剪切 | C | | 1.0 (按元素计算) |
| 下降路径 | 0.1/0.3/0.4 | 0.3 | 0.2 |
| EMA | 0.9999 | C | C |
| 标签平滑化 ϵ | 0.1 | | 0.1 |
| 数据增强 | 兰德奥格 (9, 0.5) | | 兰德奥格 (9, 0.5) |
| 混合 | 0.8 | | C |
| CutMix | 1.0 | | C |
| 随机擦除 | 0.25 | | C |

D.3 用于微调的超参数

本节给出了在 ImageNet-1K 和 downstrea COCO 对象检测和实例分割、ADE20K 语义分割任务中用于微调的超参数，如表 12、表 13 和表 14 所示。12, Tab.表 13 和表 14。14.

表 12：在 ImageNet-1K 分类中微调 RevCol 的超参数

| 超参数 | ImageNet-1K |
|-----------------|------------------------------------|
| | B/L/XL/H |
| 输入分辨率 | 384 /384 /384 /640 ²²²² |
| 微调历时 | 30 |
| 预热时间 | 0 |
| 批次大小 | 512 |
| 峰值学习率 | 5e-5 |
| 分层学习率衰减0 | |
| AdamW momentum | .9/0.8/0.8/0.8 (0.9, 0.999) |
| 权重衰减 | |
| 学习率时间表 | 余弦 |
| 头部初始刻度 | 0.001 |
| 下降路径 | 0.2/0.3/0.4/0.5 |
| EMA | C/C/C/0.9999 |
| 渐变剪切10 | .0 (标准值) |
| 标签平滑 ϵ | 0.1 |
| 数据增强 | 40 (9, 0.5) Mixup |
| CutMix | C |
| 随机擦除 | 0.25 |

表 13：使用级联掩码 R-CNN 检测器微调 RevCol 物体检测的超参数。

| 超参数 | IN-1K 预培训 | IN-22K 预培训 |
|-------------|--------------|----------------------|
| | RevCol-T/S/B | RevCol-B/L |
| 微调历元 批量大小 | | |
| 小 | | 36 16 |
| 峰值学习率 热身 | 2e-4 | 1e-4 |
| 步骤 | 0.85/0.8/0.8 | 1500 0.9/0.8 |
| 分层学习率衰减 | | (0.9, 0.999) 0.05 |
| AdamW 势头 重量 | 0.3/0.4/0.4 | 0.5/0.6 |
| 衰减 | | |
| 下降路径 | | |

表 14：使用 UperNet 细分框架对 ADE20K 语义细分进行微调的 RevCol 超参数。

| 超参数 | IN-1K 预培训 | IN-22K 预培训 |
|------------|------------------|------------------|
| | RevCol-T/S/B | RevCol-B/L |
| 输入分辨率 微调步骤 | 512 ² | 640 ² |
| 批量大小 | | 80k 16 |
| 峰值学习率 热身步骤 | | 4e-5 1500 |
| 分层学习率衰减 | 1.0 | 0.9 |
| AdamW 势头 | | (0.9, 0.999) |
| 重量衰减 | | 0.01 |
| 下降路径 | | 0.3 |

D.3.1 下行任务中的卷积核填充技巧

根据第 3.5.5 节显示的结果，较大的卷积核性能更好，尤其是在下游任务中。为了节省预训练成本，同时获得更好的性能，我们将预训练模型权重中的 3×3 小卷积核垫大，然后在检测中进行微调。

和分割任务。受 *Net2net* (Chen 等人, 2015 年) 方法的启发，我们将预先训练的卷积层中的 3×3 内核具有高斯初始化值。为了保护预训练核不受新填充值的干扰，我们将填充值初始化为均值为 0、均值为 0、均值为 0、均值为 0、均值为 0、均值为 0、均值为 0、均值为 0、均值为 0。

极小的标准偏差 ($1e-7$)。我们只在最大的模型 RevCol-H 中使用了这一技巧。在 COCO 检测任务中，我们将预训练模型中的 3×3 内核填充为 7×7 内核大小；在 ADE20k 静态分割任务中，我们将内核填充为 13×13 ，然后在相应的数据集上进行微调，以得到最终结果。一般来说，内核填充技巧可使 $_{box}$ 的 AP 提高 $0.5 \square 0.8$ ，RevCol-H 模型的 mIoU 提高 $0.7 \square 1.0$ 。

E 特征分解的可视化

在本节中，我们将展示 RevCol 与传统的序列网络不同，它可以分离具有堆叠列的特征。我们使用在 ImageNet-1K 上预先训练好的 RevCol-S 进行分析。首先，我们将每一层最后一层输出的类激活图 (CAM) 可视化。我们采用 LayerCAM (Jiang 等人, 2021 年) 技术生成带有预测类别的 CAM。图 8 显示了激活热图。随着层和列的深入，特征集中在语义更多的区域。RevCol-S 的输出是最后一列的不同层次。这些具有高级语义的特征集中在图像的不同部分和物体的整个部分，实现了任务相关特征和任务无关特征的分离。

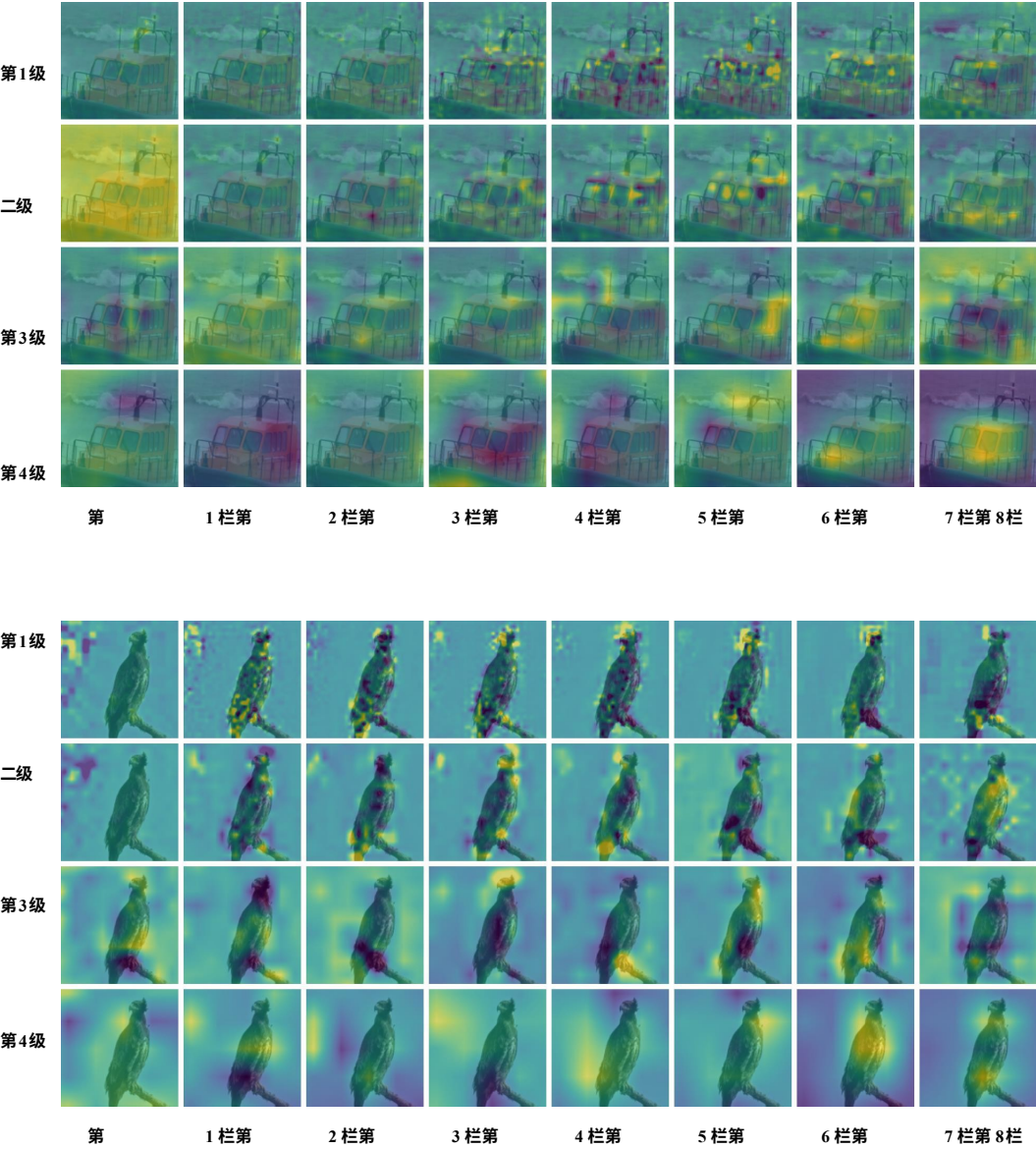


图 8：使用 LayerCAM（Jiang 等人，2021 年）对不同层和列的类激活图进行可视化。

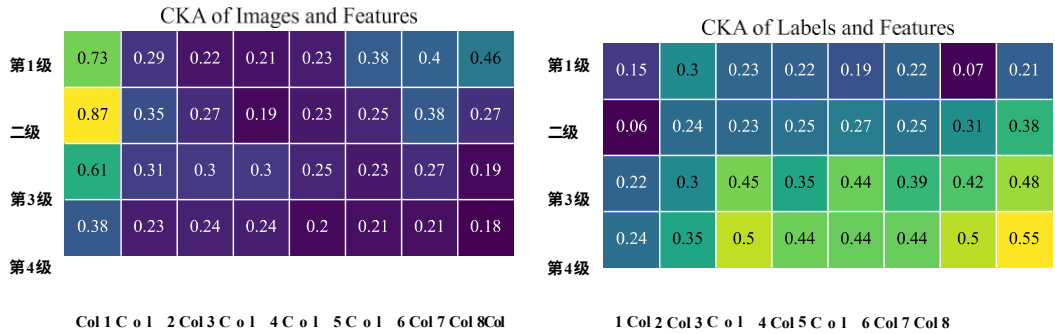


图 9：不同级别和列的特征与图像/标签的 CKA 相似度（Kornblith 等人，2019 年）。

为了量化解缠，我们使用中心核对齐 (CKA) 相似度量 (Kornblith 等人, 2019 年) 来测量 RevCol-S 中表征之间的相似性。我们计算不同层和列的中间特征与 ImageNet val set 中每个类别的图像或标签之间的 CKA 相似度。然后，我们将类别的相似性与

图 9 中的标签相似度最高。如图所示，在第 2-5 列中，图像与中间特征之间的相似性在不同级别上没有明显区别，而在第 6-8 列中，级别较高的特征与图像的相似性较低。在较高的列中，标签和中间特征之间的相似性也更加明显。