呼吸运动伪影的图像质量评估

方桂安，刘梦莎，刘玥，罗秋琳，马梓场，唐迅

摘 要—随着医学成像技术和计算机技术的不断发展和进步，医学图像分析已成为医学研究、临床疾病诊断和治疗中一个不可或缺的工具和技术手段。近几年来，深度学习（Deep learning,DL），特别是深度卷积神经网络（Convolutional neural networks,CNNs）已经迅速发展成为医学图像分析的研究热点，它能够从医学图像大数据中自动特区隐含的疾病诊断特征。本文着重于深度残差网络和 transformer 相关变种在呼吸运动伪影的图像质量评估，以此推动深度学习分类网络在病变诊断的发展。

关键词—CMR，伪影评估，深度学习，ResNet，transformer

I. 概述

CMR（心脏磁共振成像）是目前评估心脏结构和功能的金标准模式 [1]。基于机器学习的方法在以前的 CMR 比赛中取得了优异的性能（例如，ACDC [2]、M&Ms [3]）。然而，在临床实践中，模型性能受到不一致的成像环境（例如供应商和协议）、对象变化（正常人与病理病例）和意外的人类行为（例如身体运动）的挑战。在” 压力测试” 中，将经过训练的机器学习模型暴露在极端情况下，对潜在的问题进行研究是非常有意义的。

迄今为止，为了增强模型通用性，现有研究集中在供应商可变性和解剖结构变化上，而对人类行为的影响的探索较少。对于 CMR 采集，呼吸运动是主要问题之一。有急性症状的患者不能遵守屏气指令，导致图像质量下降和分析不准确。CMR 成像质量易受呼吸运动伪影的影响，具有严重呼吸运动伪影的图像不符合诊断条件，应尽可能重新获取。故评估呼吸运动伪影对 CMR 成像质量的影响，对于病变诊断具有重要意义。

方桂安，20354027，(e-mail: fanggan@mail2.sysu.edu.cn)。
刘梦莎，20354091，(e-mail: liumsh6@mail2.sysu.edu.cn)。
刘　玥，20354229，(e-mail: liuy2236@mail2.sysu.edu.cn)。
罗秋琳，20354095，(e-mail: luoqlin3@mail2.sysu.edu.cn)。
马梓场，20354103，(e-mail: mazy23@mail2.sysu.edu.cn)。
唐　迅，20354121，(e-mail: tangx66@mail2.sysu.edu.cn)。

II. 相关研究进展

基于深度学习的图像质量评估应尽可能接近主观评估，并且随着可靠技术的发展，能够最小化专家评估 MRI 的工作量，甚至从长远来看，人工智能方法完全可以替代专家。呼吸运动伪影的图像质量评估的方法可以分为完全参考（Full Reference）、简化参考（Reduced Reference）与没有参考（No Reference）方法。在完全参考和简化参考的情况下，需要首先获取无呼吸伪影的参考图像或部分区域的参考图像，将每个接收的图像的质量与参考图像进行比较，从而评估待检测图像的质量，但是在医学成像中，主要问题在于不可能获得此参考图像，否则，我们可以直接使用该参考图像进行病变诊断，无须评估图像质量，因此使用参考图像辅助评估医学图像质量是违背现实的。Chow 和 Paramesran 总结了评估医学图像质量的方法，并且预测下一代核心方法将是无参考图像的方法 [4]。因此，在该领域中，无参考图像的方法十分重要。

前人提出了许多精确评估 CMR 图像质量的方法。在 Tarroni 等人的研究中，提出了一个自动化的过程，用于确定短轴、长轴 CMR 图像的质量，研究与图像、运动伪像和对比度估计中完全心脏覆盖相关的图像质量 [5]。这项研究是在两个数据集上进行的，其中一个包含 3000 个用于培训和测试过程的样本，并根据随机森林进行 100 个样本的测试。这项研究表明，诊断全心脏覆盖范围的敏感性和特异性为 88％和 99％，诊断运动伪像 85 和 99％。在另一项类似的研究中，使用较大规模的数据集评估了相同的方法 [6]。在研究 [7] 中，研究了由于呼吸和心脏运动引起的运动伪像。在这项研究中，K 空间操作已被用来增加扭曲数据的量，并且使用 CNN 结构来学习自动检测模型。3510 张 CMR 图像的测试结果显示，ROC 曲线下方的面积（AUC）为 0.89。Zhang 等人在 5000 多个病例的数据集中研究了使用 CNN 的全左心室覆盖范围的问题，错误率低于 5％[8]。在这项研究中，将短轴 CMR 图像作为输入，使用了两

个并行的 CNN 架构来确定与心脏顶点、基础部分相关的图像的是否存在。这项研究的主要创新是在提出的体系结构的末端使用特定层来改善图像的分类，研究者引入了 Fisher 判别层，该层能够最大程度地减少类内方差，并最大程度地提高类间平均值的距离。在另一项研究 [9] 中，使用了基于 CNN 的对抗性学习方法检测心脏的左室和右心室覆盖范围，该方法已在三个数据集上进行了测试，结果表明该方法优于以前的方法。在 [10] 中，使用了生成对抗网络，以检测由 6000 多个样品组成的数据集中全左心室的覆盖范围，检测结果表明，该方法的准确性约为 90%。在 Osadebey 等人的研究中，实现了大脑和心血管图像中的图像质量评估 [11]。在这项研究中，使用了 16 卷 CMR 图像，该方法基于四种类型的添加噪声进行手工特征提取。

随着深度学习技术的发展，越来越多的研究者使用深度学习方法如卷积神经网络、注意力机制等进行图像特征的提取，并为呼吸伪影图像质量打分，因此我们探索了用于呼吸伪影图像质量评估的深度学习技术，包括残差卷积神经网络 ResNet101d，使用 CNN+Transformer 的深度网络 Coat，以及两种注意力机制的变种 Swin Transformer 与 Twins Transformer。

III. RESNET101D

深度卷积神经网络不断在图像分类任务上取得突破，网络深度的增加提升了其特征提取能力。然而随着网络深度的增加，梯度消失的问题越来越严重，网络的优化越来越困难。据此，He 等人 [12] 提出了残差卷积神经网络 (residual networks,ResNet)，进一步加深网络的同时提升了图像分类任务的性能。ResNet 由堆叠

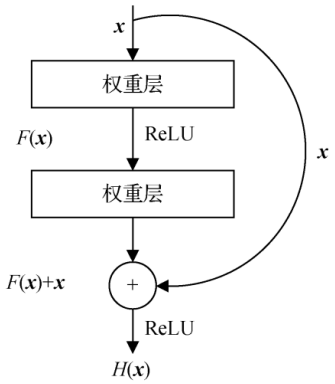


Fig. 1. 残差块

的残差块组成，残差块结构如图 1所示，残差块除了包含权重层，还通过越层连接将输入 x 直接连到输出上,$F(x)$ 为残差映射,$H(x)$ 为原始映射，残差网络令堆叠的权重层拟合残差映射 F(x) 而不是原始映射 $H(x)$，则 $F(x) = H(x) - x$，而学习残差映射较学习原始映射简单。另外，越层连接使得不同层的特征可以互相传递，一定程度上缓解了梯度消失问题。

ResNet 通过堆叠残差块使网络深度达到 152 层，残差网络在图像分类任务中获得了较大的成功。但随着网络的继续加深，梯度消失问题仍然存在，网络的优化越来越困难，为进一步提升残差网络的性能，研究者们提出了一系列残差网络的变体，本文根据这些变体基本思路的不同，将其分为 4 类: 基于深度残差网络优化的残差网络变体、采用新的训练方法的残差网络变体、基于增加宽度的残差网络变体和采用新维度的残差网络变体。基于深度残差网络优化的变体有 Pre-ResNet、加权残差网络 (weighted residual network,WResNet)、金字塔残差网络 (pyramidal residual network,PyramidalNet)、多级残差卷积神经网络 (residual networks of residual networks,RoR)、金字塔多级残差卷积神经网络 (pyramidal RoR,PRoR) 等; 采用新的训练方法有随机深度 (stochastic depth,SD) 网络、Swapout 和卷积残差记忆网络 (convolutional residual memory networks,CRMN) 等; 基于增加宽度的残差网络变体包括 ResNet in ResNet、宽残差网络 (wide residual networks,WRN) 和多残差网络 (multi-ResNet) 等，其中的宽度包括特征图中的通道数、残差块中残差函数的数量等; 最后，一些研究者在残差网络的改进中，提出了新的维度，例如基数、尺度和结构多样性等。

A. 基于深度残差网络优化的残差网络变体

He 等人 [13] 通过研究残差块中信息的传播，提出了一种新的残差单元，该残差单元去除了加法操作之后的 ReLU 激活函数，而将残差支路的操作改为 BN-ReLU-conv-BN-ReLU-conv，因此残差单元之间的信息、整个网络中的信息可以直接传播，使得网络更易训练，性能更好。

针对 ReLU 激活函数和逐元素相加之间的不兼容性，以及深度网络很难使用 MSRA 初始化器 (He 等 [14]) 使深度网络收敛等问题，Shen 等人 [15] 提出了加权残差网络 (weighted residual network,WResNet)，该网络中所有残差权重都初始化为零，并以很小的学

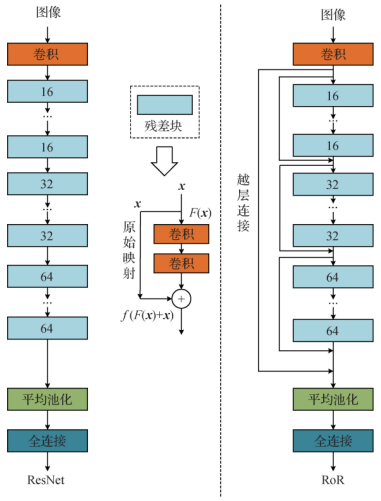


Fig. 2. RoR 模型结构 (Zhang 等人 [19])

习率 (0.001) 进行优化，这使得所有残差信号逐渐加到直接通路上，这样 1192 层的残差网络比 100 层的残差网络收敛更快，且从 100+ 层网络增长到 1 000+ 层网络，在准确率上会获得平稳的提升；最终残差权重分布在 [-0.5,0.5] 间，缓解了 ReLU 激活函数和逐元素相加之间的不兼容性。

在 ResNet 中，随着特征图通道数的增加而提取到的高级特征对分类任务更有效。Veit 等人 [16] 研究发现，在 ResNet 特征图通道数加倍的情况下，删除残差单元中带有下采样的构建块会极大降低模型分类准确率，使用随机深度训练残差网络会缓解以上情况。受此启发，Han 等人 [17] 提出了金字塔残差网络，该网络中每个残差单元的输出通道数都逐步增加，以将受下采样影响而集中分布在单个残差单元上的压力分布在所有残差单元上。Yamada 等人 [18] 将 SD-ResNet 和 PyramidalNet 合并，提出了基于分离随机深度算法的深度金字塔残差网络模型 (deep pyramidal residual networks with separated stochastic depth, PyramidalSepDrop)，该网络的残差映射 F(x) 被分成上下两部分，上部分用来增加通道数，下部分与输入 x 具有相同的通道数，两部分均使用随机深度的随机下降机制。

Zhang 等人 [19] 假设残差映射易于优化，则残差映射的残差映射更易优化，并据此在 ResNet 基础上逐级加入越层连接，构建了多级残差卷积神经网络，使得高层特征可以向低层传递，进一步抑制了梯度消失问题。RoR(residual networks for residual networks) 结构如图 2所示，首先在所有残差块外添加一个越层连接，称为一级越层连接；然后，根据卷积核滤波器的种类将残差块分为若干组，在每组残差块外添加越层连接，称为二级越层连接；随后可将每组残差块再平分，添加越层连接；最后，原始残差块中的越层连接称为末级越层连接。

RoR 网络中每组残差块的特征图尺寸和通道数保持不变，下一组残差块开始时，特征图尺寸减半、通道数加倍，这使得网络中特征信息传递不连贯，会损失一些与预测相关的有用信息，限制了网络的分类性能。针对此问题，Zhang 等人 [20] 提出了金字塔多级残差卷积神经网络，结构如图 3所示，该网络通过线性逐步增加每个残差块的输出通道数，保证高级属性多样性的同时也保证了信息的连续性。

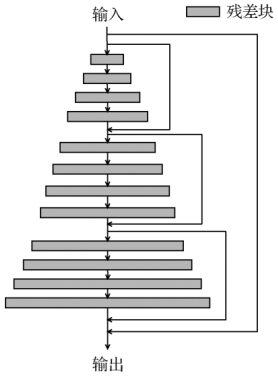


Fig. 3. 金字塔多级残差卷积神经网络 (Zhang 等 [20])

B. 采用新训练方法的残差网络变体

模型深度是模型表达能力的重要决定因素，然而非常深的网络面临着巨大的挑战: 反向传播中的梯度消失问题、前向传播中的特征重用以及网络训练耗时长等问题。针对这些问题，研究者们从训练方法角度提出了一系列残差网络的变体。Huang 等人 [21] 提出了一种新的随机深度残差网络训练方法，该方法采用了集成学习思想，在模型训练时以一定的概率随机丢弃不同的残差块，在模型训练时每次迭代训练较浅的子网络，网络测试时则采用完整的深网络，模型训练时间缩短的同时获得分类准确率的提升。Singh 等人 [22] 提出了一种新颖的随机训练方案———Swapout，其将 Dropout、随机深度等训练方法结合，从丰富的残差体系结构中取样，提高了残差网络的性能，较宽但较浅的 Swapout 网络可达到深度残差网络的性能。

Moniz 和 Pal [23] 提出了卷积残差记忆网络，该网络采用深度残差网络作为基础网络，使用长短期记忆

(long short-term memory,LSTM) 内存操作和算法架构的内存接口对网络进行训练，在 CIFAR-100 数据集上获得了当时最好的性能。

C. 基于增加宽度的残差网络变体

如上所述，为了提高残差网络的模型精度，研究者们主要致力于使 ResNet 深度更深或者深度残差网络的优化问题。而有些研究者另辟蹊径，他们提出的模型旨在使网络更宽，而不是更深，例如 ResNet in ResNet(RiR)，WRN 和 Multi-ResNet。ResNet 中的恒

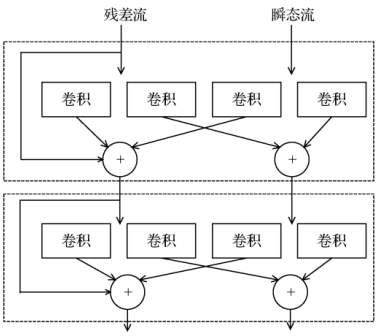


Fig. 4. 两个广义残差块

等映射导致不同特征的混合连接，然而在深度网络中，前部层学习的特征可能不再对后部层提供有用的信息。针对此问题提出了一种广义的残差结构，该广义残差结构的模块单元是由残差流和瞬态流组成的并行结构，如图 4所示，其中残差流包含越层连接且与原始残差块相似，瞬态流则是标准的卷积层，另外每个广义残差块中还有额外的两个卷积核滤波器来传递信息。两个连接的广义残差块称为 ResNet Init，将原始残差块的两个卷积层用 ResNet Init 代替，组成的新的结构称为 RiR 构建块，如图 5所示。RiR 网络在 CIFAR-10 数据集上取得了具有竞争力的结果。

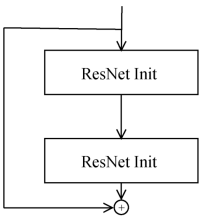


Fig. 5. 2 层的 RiR 构建块

在 ResNet 中，存在特征过度重用问题，寻求少量精确度的增加需要将网络层数加倍。针对该问题，提出了宽残差网络，WRN 在原始残差块的基础上成倍地增加残差块中卷积核的个数，增加了网络的宽度，如图 6所示，变量 k 代表宽网络卷积核较基准网络卷积核的倍数，该网络降低了网络深度，其性能远超相同层数的残差网络。实验表明，深度残差网络难以通过在整个网

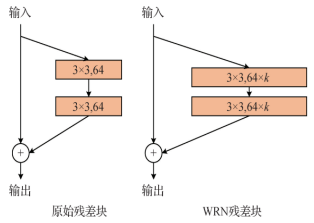


Fig. 6. 原始残差块与 WRN 残差块对比

络深度保持梯度来解决梯度消失问题，相反，可通过组合指数分布的、不同深度的网络来解决此问题，即增加网络的多重性。据此，Abdi 和 Nahavandi(2017) 提出了多残差网络 (Multi-ResNet)，该网络通过增加每个残差块中残差函数的数量，在保持深度不变的情况下，增加了网络的多样性。实验表明，增加残差块中残差函数数量比增加网络深度性能更好，与包含相同数量卷积层的深度残差网络相比，Multi-ResNet 能够取得更小的错误率。

D. 基于新维度的残差网络变体

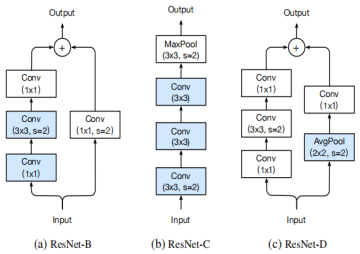


Fig. 7. 三个 ResNet 调整。ResNet-B 修改了 Resnet 的下采样块。ResNet-C 进一步修改了输入 stem。最重要的是，ResNet-D 再次修改了下采样块。

基于增加网络深度或增加网络宽度的残差网络变体都取得了很好的效果，一些研究对残差网络进行深入分析，提出了一系列基于新维度的残差网络变体。Zhang 等人 (2017) 从结构多样性的维度出发，使用 Inception 模块代替残差单元，并通过多种形式的多项式组合推广了 inception residual 单元，构建了 PolyInception 模块，这种新的设计不仅增加了结构的多样性，也增强了残差组件的表达能力。

Xie 等人 (2017) 提出了一种简单、高度模块化的网络体系结构———ResNeXt，该模型通过重复聚合一

系列具有相同拓扑结构的构建块来构建，该模型提供了一个新维度———基数，相比于增加深度和宽度，增加基数是一种更有效的获取精度提升的方法。

多尺度特征表示在分类任务中非常重要，从 AlexNet 到 ResNets，这些骨干卷积神经网络的进展不断展现出多尺度特征表示的重要作用，Gao 等人 (2021) 提出的 Res2Net 网络在更细粒度层次上提高了多尺度表征能力。该模块以瓶颈块为基础，应用了一个新的维度———尺度 (scale)，将 1×1 卷积的输出特征图在通道维度上均匀拆分为几个特征图子集，每个特征图子集进行卷积，并将得到的结果进行连接，再经过 1×1 卷积处理。Res2Net 在多个数据集上的图像分类任务中表现出更优的性能。

E. ResNet-d 模型

1) ResNet-B。这个调整首先出现在 ResNet 的 Torch 实现中，然后被多个作品采用。它改变了 ResNet 的降采样块。我们观察到，路径 A 中的卷积忽略了四分之三的输入特征图，因为它使用的内核大小为 1×1，步幅为 2。ResNet-B 切换路径 A 中前两个卷积的步幅大小，如图 7a 所示，因此它可以不忽略任何信息。因为第二次卷积的核大小为 3×3，所以路径 a 的输出形状保持不变。
2) ResNet-C。这个调整最初是在 Inception-v2 中提出的，它被应用到多个模型中，如 SENet，PSPNet，DeepLabV3，以及 ShuffleNetV2 。我们可以观察到，卷积的计算成本与内核宽度或高度的二次方成正比。7×7 卷积的成本是 3×3 卷积的 5.4 倍。因此，这个调整用三个保守的 3×3 卷积替换了输入端中的 7×7 卷积，如图 7b 所示，第一和第二卷积的输出通道为 32，步幅为 2，而最后一个卷积使用了 64 输出通道。
3) ResNet-D。降采样块路径 B 中的 1×1 卷积忽略了 3/4 的输入特征图，如果能对其进行修改，就不会忽略任何信息。实践发现，在卷积前添加一个步幅为 1 的 2×2 的平均池化层，效果良好，对计算成本的影响很小。这个调整如图 7c 所示。

IV. Swin Transformer

Swin Transformer 的最大贡献是提出了一个可以广泛应用到所有计算机视觉领域的 backbone，并且大多数在 CNN 网络中常见的超参数在 Swin Transformer 中也是可以人工调整的，例如可以调整的网络块数，每一块的层数，输入图像的大小等等。该网络架构的设计非常巧妙，是一个非常精彩地将 Transformer 应用到图像领域的结构，值得每个 AI 领域的人前去学习。

在 Swin Transformer 之前的 ViT 和 iGPT，它们都使用了小尺寸的图像作为输入，这种直接 resize 的策略无疑会损失很多信息。与它们不同的是，Swin Transformer 的输入是图像的原始尺寸，例如 ImageNet 的 $224 * 224$。另外 Swin Transformer 使用的是 CNN 中最常用的层次的网络结构，在 CNN 中一个特别重要的一点是随着网络层次的加深，节点的感受野也在不断扩大，这个特征在 Swin Transformer 中也是满足的。

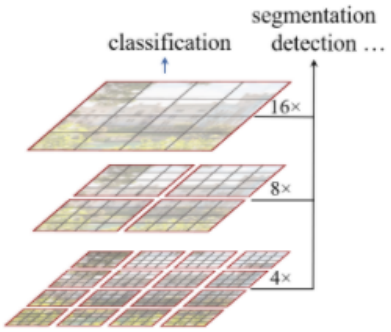


Fig. 8. Swin Transformer

A. 原理分析

Swin Transformer 提出了一种针对视觉任务的通用的 Transformer 架构 [25]，Transformer 架构在 NLP 任务中已经算得上一种通用的架构，但是如果想迁移到视觉任务中有一个比较大的困难就是处理数据的尺寸不一样。作者 [24] 分析表明，Transformer 从 NLP 迁移到 CV 上尚有缺陷主要有两点原因：

1. 最主要的原因是两个领域涉及的 scale 不同，NLP 任务以 token 为单位，scale 是标准固定的，而 CV 中基本元素的 scale 变化范围非常大。

2. CV 比起 NLP 需要更大的分辨率，而且 CV 中使用 Transformer 的计算复杂度是图像尺度的平方，这会导致计算量过于庞大，例如语义分割，需要像素级的密集预测，这对于高分辨率图像上的 Transformer 来说是难以处理的。

Swin Transformer 就是为了解决这两个问题所提出的一种通用的视觉架构。Swin Transformer 引入 CNN 中常用的层次化构建方式。

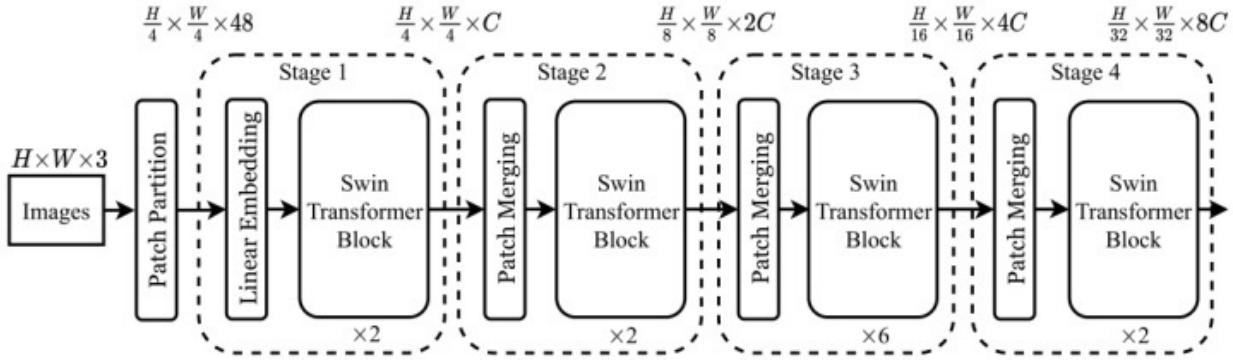


Fig. 9. Swin Transformer 架构

B. 前向传播过程

1) **图片预处理（分块和降维）**：Swin Transformer 首先把 $x \in H \times W \times 3$ 的图片，变成一个 $x_p \in N \times (P^2 \cdot C)$ 的 2 维的 image patches。它可以看做是一系列的展平的 2D 块的序列，这个序列中一共有 $N = HW/P^2$ 个展平的 2D 块，每个块的维度是 $(P^2 \cdot 3)$。其中 P 是块大小。在 Swin Transformer 中，块的大小 $P = 4$ ，所以得到的 $x_p \in N \times 48$，这里的 $N = HW/16 = \frac{H}{4} \times \frac{W}{4}$。所以经过了这一步的分块操作，一张 $x \in H \times W \times 3$ 的图片就变成了 $\frac{H}{4} \times \frac{W}{4} \times 48$ 的张量，可以理解成是 $\frac{H}{4} \times \frac{W}{4}$ 个图片块，每个块是一个 48 维的 token，如下图 9所示。

2) **线性变换**：现在得到的向量维度是: $\frac{H}{4} \times \frac{W}{4} \times 48$ ，还需要做一步叫做 Linear Embedding 的步骤，对每个向量都做一个线性变换（即全连接层），变换后的维度为 C ，这里我们称其为 Linear Embedding。这一步之后得到的张量维度是: $\frac{H}{4} \times \frac{W}{4} \times C$。

3) stage1:Swin Transformer Block: 接下来 $\frac{H}{4} \times \frac{W}{4} \times C$ 这个张量进入 2 个连续的 Swin Transformer Block 中，这被称作 Stage 1，在整个的 Stage 1 里面 token 的数量一直维持 $\frac{H}{4} \times \frac{W}{4}$ 不变。Swin Transformer Block 的结构如图 10所示。图 10是 2 个连续的 Swin Transformer Block。其中一个 Swin Transformer Block 由一个带两层 MLP 的 Shifted Window-based MSA 组成，另一个 Swin Transformer Block 由一个带两层 MLP 的 Window-based MSA 组成。在每个 MSA 模块和每个 MLP 之前使用 LayerNorm(LN) 层，并在每个 MSA 和 MLP 之后使用残差连接。

a) Window-based MSA: Window-based MSA 不同于普通的 MSA，它在一个个 window 里面去计算 self-attention。假设每个 window 里面包括 $M \times M$ 个

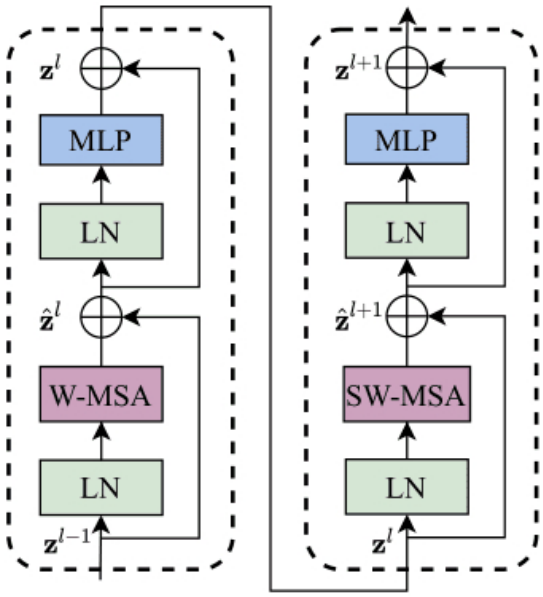


Fig. 10. Siwn Transformer Blocks

image patches，则 Window-based MSA 和普通的 MSA 的计算量分别为：

$$\begin{aligned} \Omega(\text{MSA}) &= 4hwC^2 + 2(hw)^2C \\ \Omega(\text{W}-\text{MSA}) &= 4hwC^2 + 2M^2hwC \end{aligned} \tag{1}$$

由于 Window 的 patch 数量 M 远小于图片 patch 数量 $h\omega$ ，Window-based MSA 的计算量与序列长度 $N = hw$ 成线性关系。

b) Shifted Window-based MSA: Window-based MSA 虽然大幅节约了计算量，但是牺牲了 windows 之间关系的建模，不重合的 window 之间缺乏信息交流影响了模型的表征能力。Shifted Window-based MSA 就是为了解决这个问题，如图 11所示。在两个连续的 Swin Transformer Block 中交替使用 W-MSA 和 SW-MSA。以图 10为例，将前一层 Swin Transformer Block

的 8×8 尺寸 feature map 划分成 2×2 个 patch，每个 patch 尺寸为 4×4，然后将下一层 Swin Transformer Block 的 Window 位置进行移动，得到 3×3 个不重合的 patch。移动 window 的划分方式使上一层相邻的不重合 window 之间引入连接，大大的增加了感受野。

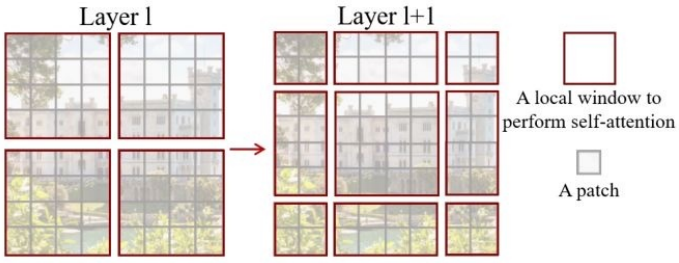


Fig. 11. Swin Transformer 架构中用于计算自我注意的移位窗口方法的示例

图 11中表示连续的 2 个 Blocks，其中第 1 个 Block 有 4 个 windows，每个 window 中是 $M\times M=4\times4$ 的 patch。第 2 个 Block 也有 4 个 windows，每个 window 中也是 $M\times M=4\times4$ 的 patch，但是 window 的位置发生了偏移，偏移的距离是 $\frac{M}{2}=2$ 这样一来，在新的 window 里面做 self-attention 操作，就可以包括原有的 windows 的边界，实现 windows 之间关系的建模。所以 2 个连续的 Swin Transformer Block 的表达式为：

$$\begin{aligned}\hat{\mathbf{z}}^{l} &= \text{W-MSA}\left(\text{LN}\left(\mathbf{z}^{l-1}\right)\right)+\mathbf{z}^{l-1},\\ \mathbf{z}^{l} &= \text{MLP}\left(\text{LN}\left(\hat{\mathbf{z}}^{l}\right)\right)+\hat{\mathbf{z}}^{l},\\ \hat{\mathbf{z}}^{l+1} &= \text{SW-MSA}\left(\text{LN}\left(\mathbf{z}^{l}\right)\right)+\mathbf{z}^{l},\\ \mathbf{z}^{l+1} &= \text{MLP}\left(\text{LN}\left(\hat{\mathbf{z}}^{l+1}\right)\right)+\hat{\mathbf{z}}^{l+1},\end{aligned}\tag{2}$$

但是引入 Shifted Window 会带来另一个问题就是会造成 window 数发生改变，而且有的 window 大，有的 window 小，比如图 12。

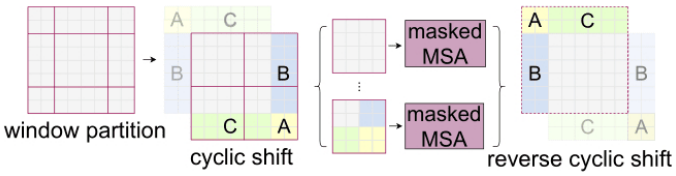


Fig. 12. 在移位窗口分区中用于自我注意的高效批处理计算方法的示例

一种简单的解决办法是把所有 window 都做 padding 操作，使之达到相同的大小。但是这会因为 window 数量的增加（从 $\lceil\frac{h}{M}\rceil\times\lceil\frac{w}{M}\rceil$ 增加到 $(\lceil\frac{h}{M}\rceil+1)\times(\lceil\frac{w}{M}\rceil+1)$）而增加计算量。所以有研究者提出了一种更加高效的 batch computation 计算方法，通过 cycle shift 的方法，合并小的 windows，仔细看图 12，将 A, B, C 这 3 个小的 windows 进行循环移位，使之合并小的 windows。经过了 cycle shift 的方法，一个 window 可能会包括来自不同 window 的内容。比如图 12右下角的 window，来自 4 个不同的 sub-window。因此，要采用 masked MSA 机制将 self-attention 的计算限制在每个子窗口内。最后通过 reverse cycle shift 的方法将每个 window 的 self-attention 结果返回。这里进行下简单的图解，下图 13代表 cycle shift 的过程，这 9 个 window 通过移位从左边移动到右侧的位置。

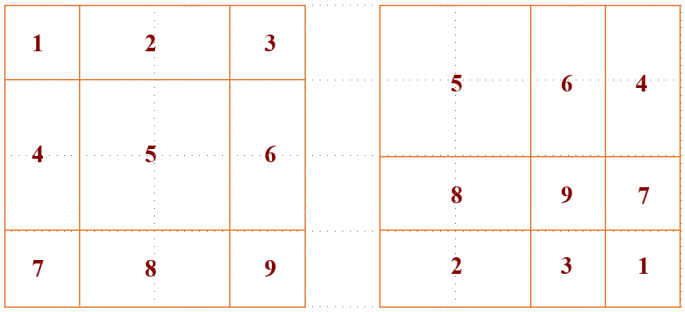


Fig. 13. cycle shift 的过程

这样再按照之前的 window 划分，就能够得到 window 5 的 attention 的结果了。

4) stage2/3/4: Stage 2 的输入是维度是 $\frac{H}{4}\times\frac{W}{4}\times C$ 的张量。从 Stage 2 到 Stage 4 的每个 stage 的初始阶段都会先做一步 Patch Merging 操作，Patch Merging 操作的目的是为了减少 tokens 的数量，它会把相邻的 2×2 个 tokens 给合并到一起，得到的 token 的维度是 4 C 。Patch Merging 操作再通过一次线性变换把维度降为 2 C 。至此，维度是 $\frac{H}{4}\times\frac{W}{4}\times C$ 的张量经过 Patch Merging 操作变成了维度是 $\frac{H}{8}\times\frac{W}{8}\times2C$ 的张量。

同理，Stage 3 的 Patch Merging 操作会把维度是 $\frac{H}{8}\times\frac{W}{8}\times2C$ 的张量变成维度是 $\frac{H}{16}\times\frac{W}{16}\times4C$ 的张量。Stage 4 的 Patch Merging 操作会把维度是 $\frac{H}{16}\times\frac{W}{16}\times4C$ 的张量变成维度是 $\frac{H}{32}\times\frac{W}{32}\times8C$ 的张量。

每个 Stage 都会改变张量的维度，形成一种层次化的表征。因此，这种层次化的表征可以方便地替换为各种视觉任务的骨干网络。

C. Swin Transformer 的结构

Swin Transformer 分为 Swin-T，Swin-S，Swin-B，Swin-L 这四种结构。使用的 window 的大小统一为 $M=7$，每个 head 的 embedding dimension 都是 32，每个 stage 的层数如下：

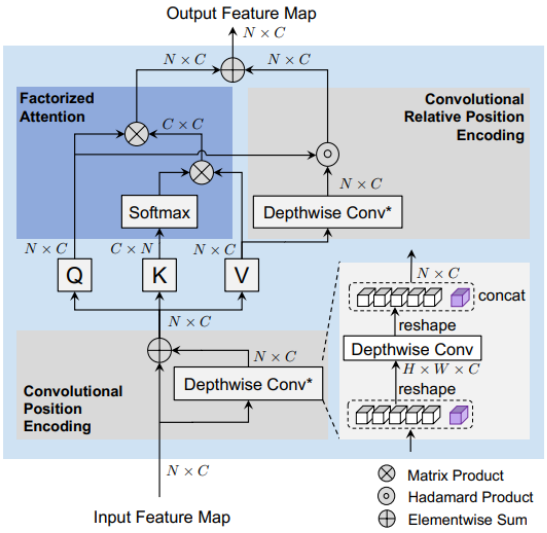


Fig. 14. 卷积注意力模块。对输入的图像标记应用卷积位置编码，得到的特征输入到具有卷积相对位置编码的因子注意力。

Swin-T: C=96 , layer number: {2,2,6,2}。
Swin-S: C=96 , layer number: {2,2,18,2}。
Swin-B: C=128 , layer number: {2,2,18,2}。
Swin-L: C=192 , layer number: {2,2,18,2}。

V. CoAT

Co-scale con-attentional transformers（CoaT），是一个基于 transformers 的图像分类器，配备了 co-scale 和 conv-attentional 机制。

首先，co-scale 机制保持了 transformers 在各个范围上编码器分支的完整性，同时允许在不同范围上学习表征能有效地相互交流；这个模型开发了两种类型的构建模块，即串行和并行模块，实现了从细到粗、从粗到细和跨尺度的图像建模。

其次，该模型设计了一个卷积注意力机制，在因子化注意力模块中实现了相对位置嵌入的表述，并采用了类似卷积的有效实现。CoaT 使图像 transformers 具有丰富的多尺度和上下文建模能力。在 ImageNet 上，与类似规模的卷积神经网络和图像 transformers 相比，相对较小的 CoaT 模型获得了卓越的分类效果。CoaT 的框架在目标检测和实例分割方面的有效性也得到了说明，显示了其对下游计算机视觉任务的适用性。

A. 卷积注意力模块

卷积注意力模块如图 14所示：我们在输入的图像标记上，应用第一个卷积位置编码。然后，我们将其送入 O ConvAtt(.)，包括因子化注意力和卷积相对位置编码。由此产生的图被用于后续的前馈网络。

因子化注意力和卷积相对位置编码的原理实现如下。

1) 因子化注意力机制: 受线性化的自注意力机制的启发，我们使用两个函数：$\phi(\cdot), \psi(\cdot) : \mathbb{R}^{N\times C} \rightarrow \mathbb{R}^{N\times C'}$ 对 softmax 注意力图进行因子化从而来近似它，并一起计算出第二个矩阵乘法（keys 和 values）。

$$\text{Factor Att}(X) = \phi(Q)\left(\psi(K)^\top V\right)$$

因式分解导致 $O\left(NC' + NC + CC'\right)$ 的空间复杂度和 $O\left(NCC'\right)$ 的时间复杂度，这两者都是序列长度为 N 的线性函数。在这里，我们按照 LambdaNets [26] 开发了我们的因子化注意力机制，将 ϕ 作为 identity 函数，将 ψ 作为 softmax。

$$\text{FactorAtt}(X) = \frac{Q}{\sqrt{C}}\left(\text{softmax}(K)^\top V\right)$$

在 LambdaNets [26] 中，缩放因子 $1/\sqrt{C}$ 被隐式地包含在权重初始化中，我们的因子化注意力应用了缩放因子。这个因子化的注意力需要 $O\left(NC + C^2\right)$ 的空间复杂度和 $O\left(NC^2\right)$ 的时间复杂度。这里需要说明的是继 [26] 之后提出的因子化注意力并不是对缩放点积注意力的直接近似，但它仍然可以被视为一种通用的注意力机制，使用 query, key and value 对特征交互进行建模。

2) 作为位置编码的卷积: 我们的因子化注意力模块减轻了原始缩放点乘注意力的编译负担。然而，由于我们首先计算 $L = \text{softmax}(K)^\top V \in \mathbb{R}^{C\times C}$, L 可以被看作是查询图 Q 中每个特征向量的全局数据依赖性线性变换。这表明，如果我们有两个 query 向量 $\mathbf{q}_1, \mathbf{q}_2 \in \mathbb{R}^C$ 来自 Q 且 $\mathbf{q}_1 = \mathbf{q}_2$，那么它们相应的自注意力输出将是相同的。如果没有位置编码，转化器将只由线性层和自我注意模块组成。因此，一个标记的输出依赖于相应的输入。这一特性对视觉任务是不利的，如语义分割（例如，天空和大海中相同的蓝色斑块被分割为同一类别）。

3) 卷积相对位置编码: 为了实现计算机视觉任务，ViT [28] 和 DeiT [29] 在输入中插入了绝对位置嵌入，这在模拟本地标记之间的关系时可能有局限性。

根据 [30]，我们可以整合一个相对位置编码 $P = \left\{\mathbf{p}_i \in \mathbb{R}^C, i = -\frac{M-1}{2}, \ldots, \frac{M-1}{2}\right\}$，窗口大小为 M，以获得相对注意力图谱 $EV \in \mathbb{R}^{N\times C}$; 在如果标记被看作是

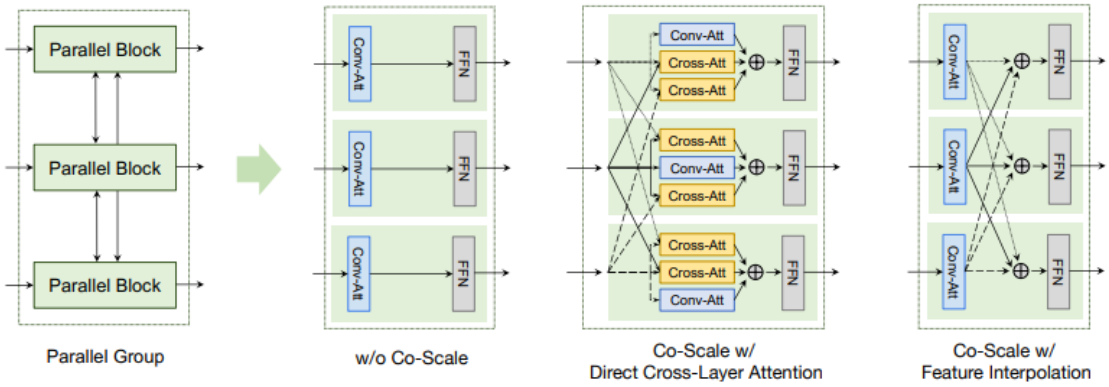


Fig. 15. CoaT 中并行块的示意图

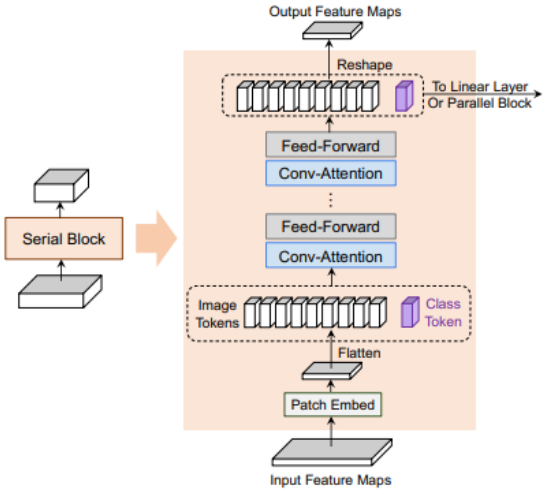


Fig. 16. CoaT 中串行块的示意图

一个一维序列，那么注意力表述如下。

$$\text{RelFactor Att}(X) = \frac{Q}{\sqrt{C}}\left(\text{softmax}(K)^\top V\right) + EV$$

其中编码矩阵 $E \in \mathbb{R}^{N\times N}$ 有以下元素。

$$E_{ij} = \mathbb{1}(i,j)\mathbf{q}_i \cdot \mathbf{p}_{j-i}, \quad 1 \le i,j \le N$$

$\mathbb{1}(i,j) = \mathbb{1}_{\{|j-i|\le(M-1)/2\}}(i,j)$ 是一个指标函数。每个元素 E_{ij} 代表从 query $\mathbf{q}_i$ 到窗口 M 内的 value $\mathbf{v}_j$，且 $(EV)_i$ 聚集了所有与 value 向量有关的 key 向量 $\mathbf{q}_i$。

然而，EV 项仍然需要 $O(N^2)$ 的空间复杂性和 $O(N^2C)$ 的时间复杂性。在 CoaT 中，我们建议将 EV 项简化为 $\hat{EV}$，即考虑每个查询中的通道。位置编码和 key 向量为通道首项。因此，对于每个通道首项 l，我们有

$$E_{ij}^{(l)} = \mathbb{1}(i,j)q_i^{(l)}p_{j-i}^{(l)}, \quad \hat{EV}_i^{(l)} = \sum_j E_{ij}^{(l)}v_j^{(l)}$$

我们可以使用一个一维深度卷积来计算 $\hat{EV}$

$$\hat{EV}V^{(l)} = Q^{(l)} \circ \text{Conv1D}\left(P^{(l)}, V^{(l)}\right)$$

$$\hat{EV} = Q \circ \text{DepthwiseConv1D}\ (P,V)$$

其中 $\circ$ 是 Hadamard 积。在视觉转换中，我们有两种类型的标记，类（CLS）标记和图像标记。因此，我们使用二维深度卷积（窗口大小为 M，内核权重为 P），并且只应用于重塑的图像标记（即 $Q^{\text{img}}, V^{\text{img}} \in \mathbb{R}^{H\times W\times C}$ from Q, V）：

$$\hat{EV}V^{\text{img}} = Q^{\text{img}} \circ \text{DepthwiseConv2D}\ \left(P, V^{\text{img}}\right)$$

$$\hat{EV} = \text{concat}\left(\hat{EV}^{\text{img}}, \mathbf{0}\right)$$

$$\text{ConvAtt}(X) = \frac{Q}{\sqrt{C}}\left(\text{softmax}(K)^\top V\right) + \hat{EV}$$

基于我们的推导，深度卷积可以被看作是相对位置编码的一个特例。

卷积相对位置编码与其他相对位置编码的比较：

通常参考的相对位置编码在标准的缩放点积注意力设置中起作用，因为编码矩阵 E 与注意力图中的 softmax logits 相结合，这在我们的因子化注意力中没有具体化。最近 LambdaNets [26] 提供了一个在资源受限情况下，深度卷积的有效变体。我们的因子化注意计算 E^V 只需要 $O(NC)$ 的空间复杂的和 $O(NCM^2)$ 的时间复杂度，具有更高效的计算结果。

4) 卷积位置编码: 接下来，我们将卷积相对位置编码的想法扩展到一般卷积位置编码的情况。卷积相对位置编码是对 queries 和 values 之间基于局部位置的关系进行建模。与大多数图像 transformers 中使用的绝对位置编码类似，我们希望将位置关系直接插入到输入图像特征中，以提升相对位置编码的效果。在每个 conv-attentional 模块中，我们在输入特征 X 中插入一

个深度卷积，并按照标准的绝对位置编码方案将产生的位置感知特征反馈给输入特征（见图 14下部），这类似于 CP VT [27] 中条件位置编码的实现。CoaT 和 CoaT-Lite 在同一区域内共享卷积位置编码权重和卷积相对位置编码权重，用于串行和并行模块。我们将卷积核大小设置为 3，用于卷积位置编码。我们将卷积核大小设置为 3、5 和 7，用于不同注意力首部的图像特征。

用于卷积相对位置编码的工作探索了卷积作为条件位置编码的使用，将其插入大小为 $\left(\frac{H}{16} \times \frac{W}{16}\right)$ 下的前馈网络之后。此模型重点是应用卷积作为相对位置编码和一般位置编码的因子化注意力机制。

B. Co-Scale Conv-Attentional Transformers (coat) 组成

所提出的 coat 机制旨在将细到粗、粗到细和跨尺度的信息引入图像 Transformers。在这里，我们描述了 CoaT 架构中的两种类型的构件，即串行和并行构件，以便对多种尺度进行建模并启用共同尺度机制。

1) CoaT Serial Block 串行块 *:* 输入的特征图首先被 Patch Embed 层下采样，然后标记化的特征（连同一个类别标记）被多个卷积层和前馈层处理。串行块（图 16）以降低的分辨率来模拟图像表示。在一个典型的串行块中，首先使用 Patch Embed 层对输入的特征图按一定的比例进行下采样，并将下采样的特征图平铺成一串图像标记。然后，我们将图像标记与一个额外的 CLS 标记连接起来，它是一个专门用来进行图像分类的向量，并应用多个卷积注意力模块来学习图像标记和 CLS 标记的内部关系。最后，我们将 CLS 标记从图像标记中分离出来，并将图像标记重塑为二维特征图，用于下一个序列块。

2) CoaT Parallel Block 并行块 *:* 我们在每个并行组中的并行块之间实现了一个共同尺度机制（如图 15所示）。在一个典型的并行组中，我们有来自不同尺度的串行块的输入特征序列（图像标记和 CLS 标记）。为了在并行组中实现跨尺度的交互，这里开发了两种策略：（1）直接跨层注意；（2）带有特征插值的注意。在本文中，我们采用了带有特征插值的注意，以获得更好的性能。接下来介绍这两种策略。

a. 直接跨层关注

在直接跨层注意中，我们从每个尺度的输入特征中得到 query, key, value 向量。对于同一层内的注意，我们使用 Conv-attention（图 14）与得到的 query, key, value 向量。对于不同层的注意，我们下取样或上取样将 key 和 value 向量矢量的分辨率与其他尺度的分辨率相匹配。然后，我们进行交叉注意，用当前尺度的 key 和 value 将另一尺度的 query 向量来扩展 conv-attention。最后，我们把 conv-attention 和 cross attention 的输出加在一起，并应用一个共享的前馈层。通过直接的跨层注意，跨尺度的信息能够以交叉注意的方式被融合在一起。

b. 带有特征插值的注意

首先，来自不同尺度的输入图像特征被独立的卷积注意模块处理。然后，我们对每个尺度的图像特征进行下采样或上采样，使其与其他尺度的尺寸相匹配，或者对自己的尺度保持相同。属于同一尺度的特征在并行组中计算总和，并进一步进入共享前馈层。通过这种方式，下一步的 conv attentional 模块可以学习跨尺度信息。

C. CoaT 模型结构

1) CoaT-Lite: 由图 17（左），按照从细到粗的金字塔结构，用一系列串行块处理输入图像。给定一个输入图像 $I \in \mathbb{R}^{H \times W \times C}$，每个序列块将图像特征向下采样为较低的分辨率，从而产生一个四种分辨率的序列:$F_1 \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times C_1}, F_2 \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times C_2}, F_3 \in \mathbb{R}^{\frac{H}{16} \times \frac{W}{16} \times C_3}, F_4 \in \mathbb{R}^{\frac{H}{32} \times \frac{W}{32} \times C_4}$ 在 CoaT-Lite 中，我们在最后一个序列块中获得 CLS 标记，并通过基于 CLS 标记的线性投影层进行图像分类。

2) CoaT: 我们的 CoaT 模型，如图 17（右），由串行和并行块组成。一旦我们从串行块中获得多尺度特征 $\{F_1, F_2, F_3, F_4\}$，我们就把 F_2, F_3, F_4 和相应的 CLS 标记传给具有三个独立并行块的并行组。

3) 模型的变体*:* 在本文中，我们探讨了 CoaT 和 CoaT-Lite 的几种不同的模型尺寸，即 Tiny, Mini, Small 和 Medium。具体来说，这些小模型有四个串行块，每个都有两个 conv-attentional 模块。在 CoaT-Lite Tiny 架构中，后面的块增加了注意力层的隐藏层。CoaT Tiny 将平行组中注意力层的隐藏尺寸设定为相等，并在六个平行组内执行共同尺度机制。Mini、small 和 med ium 模型遵循相同的结构设计，但在区块内增加了嵌入尺寸和 conv-attentional 模块的数量。

VI. Twins Transformer

Vision transformer 在视觉任务的远程依赖建模方面具有很大的灵活性，较少的诱导偏差引入使得它可以自然地处理包括图像、视频、文本、语音信号和点云在内

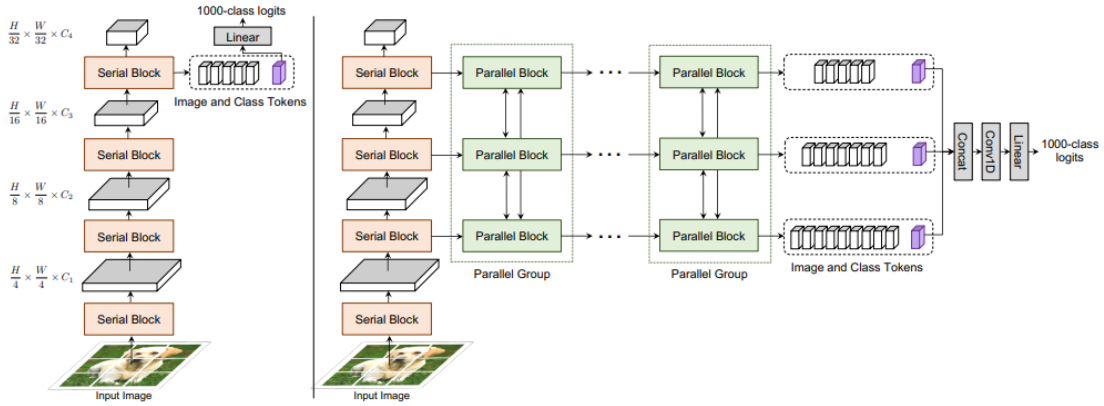


Fig. 17. （左图）CoaT-Lite 的整体网络结构。CoaT Lite 仅由串行块组成，其中图像特征被向下采样并按顺序处理。（右图）CoaT 的整体网络结构。CoaT 由串行块和并行块组成。

的多模态输入数据。本文所提出的架构在广泛的视觉任务上实现了优异的性能，包括图像级分类以及密集检测和分割。简单的结构和强大的性能证明了该架构作为许多视觉任务的更强大的骨干的可行性。将 transformer 应用于视觉任务的一个主要挑战是 transformer 的空间自注意力机制带来的巨大计算复杂度，并且以输入图像像素的数量二次增长。

A. Twins-PCPVT

针对这个问题，一种解决方法是使用局部分组的空间自注意力机制（或非重叠窗口中的空间自注意力机制（如最近提出的 Swin Transformer [33]）。传统的自注意力机制仅在每个子窗口中进行计算，而 swin 将输入在空间上分组到非重叠窗口中。虽然这种思路可以显著降低复杂性，但它缺乏不同窗口之间的连接，从而导致有限的接受域。正如许多之前的研究（[33]、[34]、[35]、[36]）所指出的，足够大的接收域对性能至关重要，特别是对于如图像分割和目标检测等密集的预测任务。针对这个问题 Swin [33] 提出了移动窗口操作，即这些局部窗口的边界随着网络的进展而逐渐移动。Swin 的这种转换后的窗口对整体性能的提升有比较理想的效果。

对于 transformer 的计算复杂度问题，PVT[8] 给出了另一种解决方案。PVT 与标准的自注意力机制（每个查询使用所有输入标记计算注意权重）不同的是，在 PVT 中每个查询只使用输入标记的下采样形式计算注意力。虽然它在理论上的计算复杂度仍然是二次的，但在实践中已经达到了相对理想的可控程度。

综上来看，Vision transformer 的核心是如何设计空间注意力。而根据研究发现，PVT 中的全局下采样

	Output Size	Layer Name	Twins-PCPVT-S	Twins-PCPVT-B	Twins-PCPVT-L
Stage 1	$\frac{H}{4} \times \frac{W}{4}$	Patch Embedding	$P_1 = 4; C_1 = 64$		
		Transformer Encoder with PEG	$\begin{bmatrix} R_1 = 8 \\ N_1 = 1 \\ E_1 = 8 \end{bmatrix} \times 3$	$\begin{bmatrix} R_1 = 8 \\ N_1 = 1 \\ E_1 = 8 \end{bmatrix} \times 3$	$\begin{bmatrix} R_1 = 8 \\ N_1 = 1 \\ E_1 = 8 \end{bmatrix} \times 3$
Stage 2	$\frac{H}{8} \times \frac{W}{8}$	Patch Embedding	$P_2 = 2; C_2 = 128$		
		Transformer Encoder with PEG	$\begin{bmatrix} R_2 = 4 \\ N_2 = 2 \\ E_2 = 8 \end{bmatrix} \times 3$	$\begin{bmatrix} R_2 = 4 \\ N_2 = 2 \\ E_2 = 8 \end{bmatrix} \times 3$	$\begin{bmatrix} R_2 = 4 \\ N_2 = 2 \\ E_2 = 8 \end{bmatrix} \times 8$
Stage 3	$\frac{H}{16} \times \frac{W}{16}$	Patch Embedding	$P_3 = 2; C_3 = 320$		
		Transformer Encoder with PEG	$\begin{bmatrix} R_3 = 2 \\ N_3 = 5 \\ E_3 = 4 \end{bmatrix} \times 6$	$\begin{bmatrix} R_3 = 2 \\ N_3 = 5 \\ E_3 = 4 \end{bmatrix} \times 18$	$\begin{bmatrix} R_3 = 2 \\ N_3 = 5 \\ E_3 = 4 \end{bmatrix} \times 27$
Stage 4	$\frac{H}{32} \times \frac{W}{32}$	Patch Embedding	$P_4 = 2; C_4 = 512$		
		Transformer Encoder with PEG	$\begin{bmatrix} R_4 = 1 \\ N_4 = 8 \\ E_4 = 4 \end{bmatrix} \times 3$	$\begin{bmatrix} R_4 = 1 \\ N_4 = 8 \\ E_4 = 4 \end{bmatrix} \times 3$	$\begin{bmatrix} R_4 = 1 \\ N_4 = 8 \\ E_4 = 4 \end{bmatrix} \times 3$

Fig. 18. Twins-PCPVT 的参数配置

注意力是非常有效的，并且通过适用的位置编码 [38]，其性能可以与最先进的视觉变压器（如 Swin）持平甚至更好。而 Twins-PCPVT 基于这一发现对 transformer 进行了重新地思考和设计（如图 19）。

1) PVT: 首先，为了更好地解决如目标检测和语义分割等的密集预测任务，PVT [37] 引入了金字塔多级设计。另外，它继承了事先设计的绝对位置编码 ViT [31] 和 DeiT [32]，各层利用全局注意机制并依靠空间缩减来降低处理整个序列的计算成本。然而，上文提到的基于局部窗口移位的 Swin transformer，在实际测试种比 PVT 性能好得多。由此研究发现，在密集的预测任务中足够大的接收域对整体性能的提升有着更关键的意义。而 PVT 性能不佳的主要原因是在 PVT[8] 中使用了绝对位置编码。具体来说，首先绝对位置编码在处理不同大小的输入时表现较差（这在密集预测任务中很常见），同时这种位置编码也打破了平移不变性。相

反，Swin 变压器利用相对位置编码，从而绕过了上述问题。

2) CPVT: CPVT [38] 使用了条件位置编码 (CPE) 来代替绝对位置编码（pvt 中的 PE 是基于输入的，因此可以自然地避免上述绝对编码的问题）。生成 CPE 的位置编码生成器 (PEG)[9] 被放置在每一级的第一个编码器块之后。CPVT 使用 PEG 最简单的形式，即不需要批处理归一化的 2D 深度卷积。对于图像级分类，在 CPVT 之后，CPVT 删除类标记，并在阶段 [38] 的末尾使用全局平均池 (GAP)；对于其他视觉任务，CPVT 遵循 PVT 的设计。

在理论上，Twins-PCPVT 继承了 PVT 和 CPVT 的优点，使其易于高效实现。另外该相关工作的实验结果也表明，这种简单的设计完全可以媲美当前非常先进的 Swin transformer 的性能。图 18展示了 Twins-PCPVT 的详细结构设计，值得关注的是 Twins-PCPVT 具有与 [37] 相似的 FLOPs 和参数数量，这也是其充分继承 PVT[8] 优点的结构依据。

B. Twins-SVT

在密集的预测任务中，由于高分辨率输入，视觉转换器的计算量非常大。假设输入分辨率为 $H\times W$，维数为 d 的自我注意复杂度为 $O(H^2W^2d)$。下面介绍空间可分离的自我注意 (SSSA)。SSSA 由局部分组的 LSA (local -grouped self-attention) 和全局下采样的 GSA (global sub-sampled attention) 组成。

1) Locally-grouped self-attention (LSA): 在深度卷积组设计的基础上，LSA 首先将二维特征映射平均划分为为 m × n 个子窗口，使自注意力通信只发生在每个子窗口内。

在不丧失一般性的情况下，我们假设每组包含 HW/mn 个元素，同时 $H\%m=0$、$W\%n=0$。可以得出在这个窗口中自注意力的计算代价是 $O(\frac{H^2W^2}{m^2n^2}d)$，总的代价函数是 $O(\frac{H^2W^2}{mn}d)$。如果令 $k1=Hm$、$k2=Wn$，成本可以计算为 $O(k_1K_2HWd)$。虽然局部分组的自注意力机制是易于计算的，但图像已经被划分为了不重叠的子窗口。因此，我们需要一种像在 Swin 中一样的机制（详见上文）来在不同的子窗口之间进行通信。否则，信息将被限制在局部处理，从而使得接收野很小并显著降低性能。

2) Global sub-sampled attention (GSA): 针对接受野过小的问题，一个简单的解决方案是在每个局部注意块之后添加额外的标准全局自我注意层，从而实现跨组信息交换。然而，这种方法会带来 $O(H^2W^2d)$ 的计算复杂度。这里，GSA 使用一个代表总结 m × n 个子窗口的重要信息，并使用该代表与其他子窗口进行通信，这可以大大降低成本到

$$\mathcal{O}(mnHWd)=\mathcal{O}\left(\frac{H^2W^2d}{k_1k_2}\right) \tag{3}$$

这在本质上相当于使用下采样特征映射作为自注意操作，因此被称之为全局下采样注意 (GSA)。如果交替使用前面提到的 LSA 和 GSA，就类似于可分离卷积 (深度方向 + 点方向)，此时计算总代价为

$$\mathcal{O}\left(\frac{H^2W^2d}{k_1k_2}+k_1k_2HWd\right) \tag{4}$$

另外我们有

$$\frac{H^2W^2d}{k_1k_2}+k_1k_2HWd\geq 2HWd\sqrt{HW} \tag{5}$$

解不等式得最小值是在 $k_1\cdot k_2=\sqrt{HW}$ 处获得。在不失一般性的前提下，当使用正方形子窗口（即 k1 = k2）且当 H = W = 224（该取值在图像分类任务中被普遍使用）时，k1 = k2 = 15 时结果接近全局最小值。然而 GSA 的网络被设计为包括几个不同分辨率的阶段。阶段 1 的特征图为 56 × 56, $k1=k2=\sqrt{56}$ 时最小。理论上，我们可以为每个阶段校准最优 k1 和 k2。为了简单起见，GSA 在任何地方都使用 k1 = k2 = 7。对于分辨率较低的阶段，控制 GSA 的汇总窗口大小以避免生成的密钥数量过少。具体实施时 GSA 对最后三个阶段分别取值为 4、2 和 1。该空间可分离自我注意力 (SSSA) 可以写成如下形式：

$$\hat{\mathbf{z}}_{ij}^{l}=\text{LSA}\left(\text{LayerNorm}\left(\mathbf{z}_{ij}^{l-1}\right)\right)+\mathbf{z}_{ij}^{l-1} \tag{6}$$

$$\mathbf{z}_{ij}^{l}=\text{FFN}\left(\text{LayerNorm}\left(\hat{\mathbf{z}}_{ij}^{l}\right)\right)+\hat{\mathbf{z}}_{ij}^{l} \tag{7}$$

$$\hat{\mathbf{z}}^{l+1}=\text{GSA}\left(\text{LayerNorm}\left(\mathbf{z}^{l}\right)\right)+\mathbf{z}^{l} \tag{8}$$

$$\mathbf{z}^{l+1}=\text{FFN}\left(\text{LayerNorm}\left(\hat{\mathbf{z}}^{l+1}\right)\right)+\hat{\mathbf{z}}^{l+1} \tag{9}$$

$i\in\{1,2,\ldots m\}, j\in\{1,2,\ldots,n\}$

这里的 LSA 是指子窗口内的本地分组自我注意；GSA 是通过与每个子窗口的代表键 (由子采样函数生成) 交互的全局子采样注意力。和标准的 self-attention 一样，LSA 和 GSA 都有多个头部。LSA 的 PyTorch

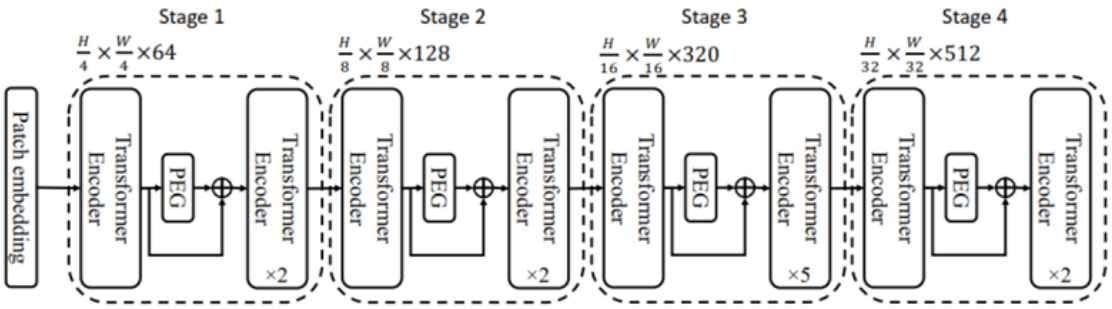


Fig. 19. Twins-PCPVT 的整体架构

Algorithm 1 PyTorch snippet of LSA.

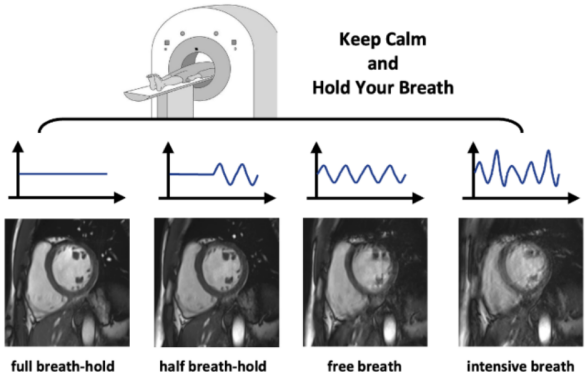
```
class GroupAttention(nn.Module):
    def __init__(self, dim, num_heads=8, qkv_bias=False, qk_scale=None, attn_drop=0., proj_drop=0., k1=7,
                 k2=7):
        super(GroupAttention, self).__init__()
        self.dim = dim
        self.num_heads = num_heads
        head_dim = dim // num_heads
        self.scale = qk_scale or head_dim ** -0.5
        self.qkv = nn.Linear(dim, dim * 3, bias=qkv_bias)
        self.attn_drop = nn.Dropout(attn_drop)
        self.proj = nn.Linear(dim, dim)
        self.proj_drop = nn.Dropout(proj_drop)
        self.k1 = k1
        self.k2 = k2

    def forward(self, x, H, W):
        B, N, C = x.shape
        h_group, w_group = H // self.k1, W // self.k2
        total_groups = h_group * w_group
        x = x.reshape(B, h_group, self.k1, w_group, self.k2, C).transpose(2, 3)
        qkv = self.qkv(x).reshape(B, total_groups, -1, 3, self.num_heads, C // self.num_heads).permute(3, 0,
                1, 4, 2, 5)
        q, k, v = qkv[0], qkv[1], qkv[2]
        attn = (q @ k.transpose(-2, -1)) * self.scale
        attn = attn.softmax(dim=-1)
        attn = self.attn_drop(attn)
        attn = (attn @ v).transpose(2, 3).reshape(B, h_group, w_group, self.k1, self.k2, C)
        x = attn.transpose(2, 3).reshape(B, N, C)
        x = self.proj(x)
        x = self.proj_drop(x)
        return x
```

Fig. 20. LSA 的 Pytorch 代码展示

	Output Size	Layer Name	Twins-SVT-S	Twins-SVT-B	Twins-SVT-L
Stage 1	$\frac{H}{4} \times \frac{W}{4}$	Patch Embedding	$P_1 = 4; C_1 = 64$	$P_1 = 4; C_1 = 96$	$P_1 = 4; C_1 = 128$
		Transformer Encoder w/ PEG	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 1$	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 1$	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 1$
Stage 2	$\frac{H}{8} \times \frac{W}{8}$	Patch Embedding	$P_2 = 2; C_2 = 128$	$P_2 = 2; C_2 = 192$	$P_2 = 2; C_2 = 256$
		Transformer Encoder w/ PEG	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 1$	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 1$	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 1$
Stage 3	$\frac{H}{16} \times \frac{W}{16}$	Patch Embedding	$P_3 = 2; C_3 = 256$	$P_3 = 2; C_3 = 384$	$P_3 = 2; C_3 = 512$
		Transformer Encoder w/ PEG	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 5$	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 9$	$\begin{bmatrix} LSA \\ GSA \end{bmatrix} \times 9$
Stage 4	$\frac{H}{32} \times \frac{W}{32}$	Patch Embedding	$P_4 = 2; C_4 = 512$	$P_4 = 2; C_4 = 768$	$P_4 = 2; C_4 = 1024$
		Transformer Encoder w/ PEG	$\begin{bmatrix} GSA \end{bmatrix} \times 4$	$\begin{bmatrix} GSA \end{bmatrix} \times 2$	$\begin{bmatrix} GSA \end{bmatrix} \times 2$

Fig. 21. Twins-SVT 的参数配置



代码在图 20中给出。同样，这里使用 CPVT[9] 的 PEG 来编码位置信息并把它插入到每个阶段的第一个块之后，实现动态地处理可变长度的输入。

Twins - SVT 的详细配置如图 21所示。这里尽量使用 Swin [33] 中类似的设置，以确保良好的性能是由于新的设计范式。

VII. 结论

A. 数据集简介

在本次研究中，我们用到的是 CMRxMotion 数据集，这是一个真实的心脏 MRI 数据集，包括具有不同呼吸运动水平的极端情况。我们的目标是通过这样一个极端的 CMR 数据集，以便更全面地评估呼吸运动下的模型鲁棒性。

收集者通过重新设计参与者在获取过程中的行为来攻击程序标准 (SOP)。对于 45 名健康志愿者，使用相同的 MRI 扫描仪（西门子 3T MRI 扫描仪 MAGNETOM Vida）进行临床 CMR 扫描。志愿者被训练以 4 种方式行动，分别为：

1) 遵守屏气指令，
2) 将屏气时间减半，
3) 自由呼吸，
4) 集中呼吸。

因此，对于单个志愿者，收集者在 4 个呼吸运动水平下获得了一组成对的 CMR 图像。放射科医生首先评

5-point score	Grade	解释
5	诊断质量极佳	不存在人工制品
4	在足以诊断以上	存在轻微伪影，但图像质量有所降低
3	足以诊断	存在轻微伪影，图像质量有所降低，但仍足以进行诊断
2	诊断有问题	图像质量因伪影而受损，因此图像的诊断价值值得怀疑
1	非诊断性	图像质量因伪影而无法评估，严重受损

估图像质量并识别质量差的图像。对于具有诊断质量的图像，放射科医生进一步分割左心室、左心室心肌和右心室。该数据集发布了 25 名志愿者的图像、质量分数和真实分割（如果有的话），用于训练和验证。其余 20 名志愿者将被留作参加测试。

所有图像都在 3D Slicer 中查看，图像质量由放射科医师评分。标准的李克特 5 点量表使用如下：

- 诊断质量极佳（5），
- 在足以诊断以上（4），
- 足够诊断（3），
- 诊断有问题（2），
- 非诊断性（1）。

为了获得更好的再现性，根据原始的 5 分分数定义了 3 个级别的运动伪影。质量得分为 4-5 的图像被标记为轻度运动伪影，质量得分为 3 的图像被标记为中等运动伪影，质量得分为 1-2 的图像被标记为严重的运动伪影。

值得注意的是，预处理时 CMR 图像被匿名化并从 DICOM 文件导出为 NIFTI 文件，故不同于普通的 png/jpg 等格式可以直接被模型读取并训练，需要使用 python 中的 nibabel 分割。nii 图片数据是三维的，读入之后根据 label 再沿 x，y，z 某一方向保存为 png，即可正常训练。

B. 结果

训练结果如表 1 所示，在同等训练配置下，最终 Coat 取得了最优的结果。

综上所述，深度学习具有自动地从数据中学习深层次、更具鉴别性特征的能力，已应用于医学图像分析的多个研究领域，并取得了突破性进展。我们注意到，在大多数文献中，使用深度学习相关方法展示了其领先水平的性能，这已由医学图像分析的若干计算挑战赛结果证明；其次，云计算和多 GPU 高性能并行计算技术的发展，使得深度学习从海量的医学图像大数据中学习深层特征成为可能；最后，可公开访问的相关医学图像数据库的出现及多个医学图像分割挑战赛数据集，使得基于深度学习的医学诊断能够得到有效验证。

表 1 准确率

编号	模型	Train_acc	Valid_acc
1	ResNet101d	70.1%	72.8%
2	swin_base_patch4_window7_224	82.1%	68.6%
3	coat_lite_small	82.4%	78.6%
4	twins_pcpvt_base	62.4%	66.4%
5	twins_svt_base	63.5%	66.3%

我们相信，通过深度学习算法的不断改进，借助高性能并行计算技术的发展和日益改善的医学图像质量与不断增长的医学图像标记样本集，基于深度学习的医学图像分析将大有所为；随着基于深度学习的图像质量评估研究不断深入，接近甚至成功专家水平的智能诊断终将实现！

参考文献

[1] Schulz-Menger, Jeanette, et al. ”Standardized image interpretation and post-processing in cardiovascular magnetic resonance-2020 update.” Journal of Cardiovascular Magnetic Resonance 22.1 (2020): 1-22.

[2] Bernard, Olivier, et al. ”Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved?.” IEEE Transactions on Medical Imaging 37.11 (2018): 2514-2525.

[3] Campello, Víctor M., et al. ”Multi-centre, multi-vendor and multi-disease cardiac segmentation: the M&Ms challenge.” IEEE Transactions on Medical Imaging 40.12 (2021): 3543-3554.

[4] Chow, L. S., & Paramesran, R. (2016). Review of medical image quality assessment. Biomedical signal processing and control, 27, 145-154.

[5] Tarroni, G., Oktay, O., Bai, W., Schuh, A., Suzuki, H., Passerat-Palmbach, J., ... & Rueckert, D. (2018). Learning-based quality control for cardiac MR images. IEEE transactions on medical imaging, 38(5), 1127-1138.

[6] Tarroni, G., Bai, W., Oktay, O., Schuh, A., Suzuki, H., Glocker, B., ... & Rueckert, D. (2020). Large-scale quality control of cardiac imaging in population studies: application to UK Biobank. Scientific reports, 10(1), 1-11.

[7] Oksuz, I., Ruijsink, B., Puyol-Antón, E., Clough, J. R., Cruz, G., Bustin, A., ... & King, A. P. (2019). Automatic CNN-based detection of cardiac MR motion artefacts using k-space data augmentation and curriculum learning. Medical image analysis, 55, 136-147.

[8] Zhang, L., Gooya, A., Pereanez, M., Dong, B., Piechnik, S. K., Neubauer, S., ... & Frangi, A. F. (2018). Automatic assessment of full left ventricular coverage in cardiac cine magnetic resonance

imaging with fisher-discriminative 3-D CNN. IEEE Transactions on Biomedical Engineering, 66(7), 1975-1986.

[9] Zhang, L., Pereañez, M., Piechnik, S. K., Neubauer, S., Petersen, S. E., & Frangi, A. F. (2018, September). Multi-input and dataset-invariant adversarial learning (MDAL) for left and right-ventricular coverage estimation in cardiac MRI. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 481-489). Springer, Cham.

[10] Zhang, L., Gooya, A., & Frangi, A. F. (2017, September). Semi-supervised assessment of incomplete LV coverage in cardiac MRI using generative adversarial nets. In International workshop on simulation and synthesis in medical imaging (pp. 61-68). Springer, Cham.

[11] Osadebey, M., Pedersen, M., Arnold, D., & Wendel-Mitoraj, K. (2018). Image quality evaluation in clinical research: A case study on brain and cardiac MRI images in multi-center clinical trials. IEEE journal of translational engineering in health and medicine, 6, 1-15. doi: 10.1109/CVPR.2016.90.

[12] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[13] He, K., Zhang, X., Ren, S., Sun, J. (2016). Identity Mappings in Deep Residual Networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision –ECCV 2016. ECCV 2016. Lecture Notes in Computer Science(), vol 9908. Springer, Cham. https://doi.org/10.1007/978-3-319-46493-0_38

[14] K. He, X. Zhang, S. Ren and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1026-1034, doi: 10.1109/ICCV.2015.123.

[15] F. Shen, R. Gan and G. Zeng, "Weighted residuals for very deep networks," 2016 3rd International Conference on Systems and Informatics (ICSAI), 2016, pp. 936-941, doi: 10.1109/ICSAI.2016.7811085.

[16] Veit A, Wilber M, Belongie S. Residual Networks Behave Like Ensembles of Relatively Shallow Networks[J]. Advances in Neural Information Processing Systems, 2016.

[17] Han D, Kim J, Kim J. Deep Pyramidal Residual Networks[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.

[18] Yamada Y, Iwamura M, Kise K. Deep Pyramidal Residual Networks with Separated Stochastic Depth[J]. 2016.

[19] Zhang K, Sun M, Han X, et al.Residual Networks of Residual Networks: Multilevel Residual Networks[J]. IEEE Transactions on Circuits & Systems for Video Technology, 2016:1-1.

[20] Zhang, K., Guo, L., Gao, C. et al. Pyramidal RoR for image classification. Cluster Comput 22, 5115–5125 (2019). https://doi.org/10.1007/s10586-017-1443-x

[21] Huang, G., Sun, Y., Liu, Z., Sedra, D., Weinberger, K.Q. (2016). Deep Networks with Stochastic Depth. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision –ECCV 2016. ECCV 2016. Lecture Notes in Computer Science(), vol 9908. Springer, Cham. https://doi.org/10.1007/978-3-319-46493-0_39

[22] Singh, S., Hoiem, D., & Forsyth, D. (2016). Swapout: Learning an ensemble of deep architectures. Advances in Neural Information Processing Systems, 28-36.

[23] Liang, D. et al. (2018). Residual Convolutional Neural Networks with Global and Local Pathways for Classification of Focal Liver Lesions. In: Geng, X., Kang, BH. (eds) PRICAI 2018: Trends in Artificial Intelligence. PRICAI 2018. Lecture Notes in Computer Science(), vol 11012. Springer, Cham. https://doi.org/10.1007/978-3-319-97304-3_47

[24] Liu, Z. , Lin, Y. , Cao, Y. , Hu, H. , Wei, Y. , & Zhang, Z. , et al. (2021). Swin Transformer: Hierarchical Vision Transformer using Shifted Windows.

[25] Ashish V aswani, Noam Shazeer, Niki Parmar, Jakob Uszko-reit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems, pages 5998–6008, 2017. 1,2,4

[26] Irwan Bello. Lambdanetworks: Modeling long-range interactions without attention. In ICLR, 2021.

[27] Xiangxiang Chu, Bo Zhang, Zhi Tian, Xiaolin Wei, and Huaxia Xia. Conditional positional encodings for vision transformers. arXiv preprint arXiv:2102.10882, 2021.

[28] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In ICLR, 2021.

[29] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve J ´ egou. Training ´ data-efficient image transformers & distillation through attention. arXiv preprint arXiv:2012.12877, 2020

[30] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In NAACLHLT, 2018.

[31] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint arXiv:1607.06450,2016.

[32] Irwan Bello. Lambdanetworks: Modeling long-range interactions without attention. In Proc. Int. Conf. Learn. Representations, 2021.

[33] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In Proc. IEEE Conf.Comp. Vis. Patt. Recogn., 2018.

[34] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-toend object detection with transformers. In Proc. Eur. Conf. Comp. Vis., 2020.

[35] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155, 2019.

[36] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal.Mach. Intell., 2017.

[37] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proc. Eur. Conf. Comp. Vis., 2018.

[38] Yunpeng Chen, Jianan Li, Huaxin Xiao, Xiaojie Jin, Shuicheng Yan, and Jiashi Feng. Dual path networks. Proc. Advances in Neural Inf. Process. Syst., 2017.