arXiv:2303.04599v1 [cs.CV] 8 Mar 2023

通过特征空间聚类，使用基于内容的Transformer进行点云分类

刘亚辉、田斌、吕宜生、李灵犀和王飞跃

摘要—近年来，已有一些研究尝试将 Transformer 应用于三维点云分类。为降低计算量，现有方法大多专注于局部空间注意力，却忽略了点的内容，未能建立相距较远但彼此相关的点之间的关系。为克服局部空间注意力的局限性，我们提出了一种基于点内容的Transformer架构，简称 PointConT。该架构利用点在特征空间中的局部性（基于内容），将特征相似的采样点聚为同一类，并在每一类内计算自注意力，从而有效平衡长程依赖建模与计算复杂度。我们进一步为点云分类引入了一种 Inception特征聚合器，利用并行结构分别聚合各分支中的高频和低频信息。大量实验表明，PointConT模型在点云形状分类任务中取得了出色的性能。尤其是在 ScanObjectNN 最具挑战性的设置下，我们的方法达到了 90.3% 的 Top-1准确率。本文的源代码可在 https://github.com/yahuiliu99/PointConT。

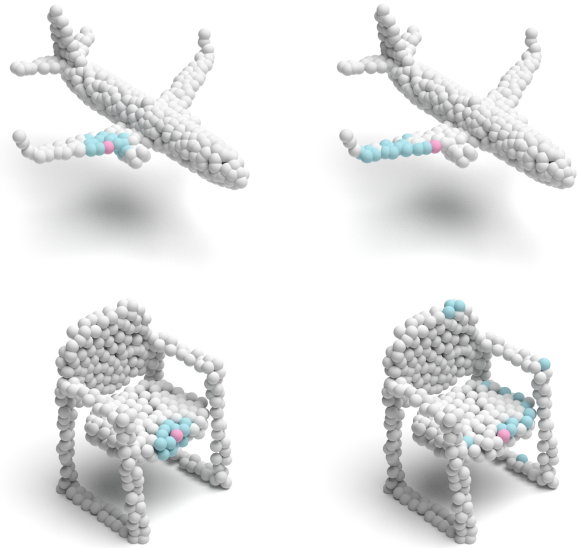
索引词—点云分类、局部注意力、基于内容的Transformer、特征聚合器、深度学习。

I. 引言

3D 点云分析在许多领域都广受关注，包括场景理解 [1]–[3],机器人学和自动驾驶车辆 [4]–[6]。与二维图像相比，三维点云能够提供充分的空间和几何信息，但其排列没有任何特定顺序。由于结构不规则，卷积神经网络无法直接应用于点云处理，而 Transformer [7] 架构本质上具有置换不变性，因此天然适用于点云学习。

近年来，研究者开始探索 Transformer 架构在点云分析中的应用 [8]–[14]。然而，这些模型普遍存在计算成本高的问题，这引起了研究者的关注，并促使他们考虑准确率与推理速度之间的权衡。降低计算复杂度的两种主要方法是点下采样和局部自注意力[8], [11]–[13]。最远点采样（FPS）等点下采样算法 [15], 能够均匀覆盖整个点云。局部自注意力则计算一个点子集（补丁或立方窗口）内点之间的关系

Y. H. Liu、B. Tian、Y. S. Lv 和 F.-Y. Wang 任职于中国北京的中国科学院自动化研究所。（电子邮箱：liuyahui2021@ia.ac.cn；bin.tian@ia.ac.cn；yisheng.lv@ia.ac.cn；feiyue@ieee.org）。L. X. Li 任职于美国印第安纳波利斯的印第安纳大学—普渡大学印第安纳波利斯联合分校（IUPUI）普渡工程与技术学院。（电子邮箱：LL7@iupui.edu）。



(a) 三维空间中的 k近邻 (b) 特征空间中的聚类（在三维空间中可视化）

图1. 三维空间局部性与基于内容的局部性的对比。红色点表示采样中心点，蓝色点表示邻域或簇。在基于内容的注意力中，点会根据其特征相似度聚类为多个簇。

该子集是在 3D 空间中划分得到的。尽管局部空间注意力显著提高了效率，但在捕捉相距较远却彼此相似的点之间的交互方面仍面临困难。

因此，我们提出了一种简单而强大的点云分类架构，称为点内容式Transformer（PointConT），**它在特征空间（基于内容）而非三维空间中利用局部自注意力**，如图1所示。我们从点云的内容出发，根据相似性将采样点聚类，并在每个类别内计算自注意力。这既保留了全局自注意力机制捕获长程特征依赖的能力，又显著降低了计算复杂度。具体而言，每个块都会根据查询的内容（即特征）动态地将所有查询划分为多个簇，并选择相应的键和值来计算局部自注意力。在Transformer的不同阶段和不同注意力头中，聚类方式会相应变化，从而充分反映内容的动态变化。需要注意的是，与k近邻（kNN）不同，这些簇彼此不重叠，因此能进一步降低计算复杂度。

此外，我们还从频率角度补充了点云特征聚合。近期研究 [16],[17] 发现，最大池化会增强高频特征，而平均池化和Transformer会削弱高频分量，这也与消融实验中的观察结果一致。为聚合高频和低频特征，我们设计了一种由两个分支组成的Inception特征聚合器，其名称“Inception”源自Inception模块 [18], [19]。高频聚合分支由最大池化操作和残差多层感知机模块组成，而低频聚合分支则通过平均池化操作和基于内容的Transformer块实现。

本文的主要贡献总结如下。

- 我们提出了点内容Transformer（Point- ConT），根据点的内容对其进行聚类，并在每个聚类内计算自注意力，从而建立长程特征依赖，同时显著减少计算量。
- 我们为点云分类设计了一种Inception特征聚合器，利用并行结构分别聚合每个分支中的高频和低频信息。
- 实验表明，我们的模型在ModelNet40 [20] 和ScanObjectNN [21] 数据集上具有竞争力。大量消融研究验证了PointConT设计中各个组件的优势。

II. 相关工作

A. 点云处理

点云处理方法主要分为两类。一类是将点云转换为规则网格结构，以便卷积神经网络直接处理，例如体素表示 [22]–[24]（通过体素化）或图像 [25], [26] （通过投影或渲染）。另一类是基于点的建模，即直接将原始点云输入深度网络，而不进行任何转换。本文聚焦于基于点的方法。

PointNet 是一项开创性工作，成功地将深度架构应用于原始点集 [27]。它采用共享的多层感知机（MLP）和最大池化构建对称函数，从而保证了置换不变性。然而，PointNet 只能学习单点特征或全局特征，因此难以捕捉点之间的相互作用。PointNet++ 建立在 PointNet 之上，能够学习层次化点云特征，并通过集合抽象 [15]聚合局部几何邻域中的特征。

在这些工作之后，一些研究将基于点的方法扩展到了各种局部聚合算子。对局部聚合算子的探索可分为三组：基于卷积的 [28]–[35], 、基于图的[36]–[39], 以及基于注意力的 [8], [9], [40]–[42] 方法。

基于卷积的方法：[31]和 [32] 通过预定义的几何先验，在局部区域内学习卷积核。另一种点卷积方法 KPConv [34], 将权重矩阵与三维空间中预定义的核点相关联。然而，固定的核点可能并不适合对复杂的三维位置变化进行建模。PAConv 构建了一个位置自适应卷积算子，该算子使用动态核 [35],在权重库中组装基本权重矩阵。组装系数通过多层感知机根据相对点位置进行学习。

基于图的方法： 基于图的方法兴起于 DGCNN [37], ，该方法在每一层动态更新的图上进行学习。它提出了一种名为 EdgeConv 的局部特征聚合算子，用于生成描述特征空间中关键点与其邻居点之间语义关系的边特征。此外，CurveNet 通过执行引导式游走，将连续片段的点分组为曲线，从而探索几何信息 [39]。

基于注意力的方法： 点云 Transformer 设计了偏移注意力以提取全局特征，并采用邻居嵌入策略来增强局部特征表示 [9]。Point Transformer 提出了一种改进的 Transformer 架构，利用向量注意力和相对位置编码聚合局部特征 [8]。受 Swin Transformer [43],启发，分层 Transformer [12], 将三维空间划分为互不重叠的立方窗口，并提出了一种用于采样键的分层策略。

此外，PointASNL [44]利用非局部网络[45]和自适应采样模块，增强长程依赖关系的学习。近期，PointNeXt [46] 在 PointNet++骨干网络的基础上探索了更先进的训练和数据增强策略，以进一步提升准确率和效率。

B. 视觉 *Transformer*

近年来，相较于人们熟悉的卷积网络，Transformer 架构在二维图像理解方面取得了巨大成功。视觉 Transformer（ViT） [47] 是首篇成功将 Transformer 编码器应用于图像的论文。它将图像划分为互不重叠的图像块（词元），然后对其进行线性嵌入。此外，金字塔视觉 Transformer（PVT） [48], [49]将层次结构引入 Transformer 框架。Transformer 中的 Transformer（TNT）[50] 在 ViT 基线模型的基础上，扩展了图像块内的子图像块级注意力。近期， [43], [51]–[53]的方法在局部窗口内计算注意力。Swin [43] 是一种代表性方法，它采用两个关键概念来改进原始 ViT——层次化特征图和移位窗口注意力。除了图像空间手工设计窗口之外，DGT [54] 和 BOAT [55] 还利用特征空间局部性。DGT [54] 动态地将查询划分为多个组，并为每组选择最相关的键/值来计算注意力。BOAT [55] 在现有的基于窗口的局部注意力基础上补充了特征空间局部注意力模块，从而显著增强了对长程特征依赖的建模能力。

尽管 Transformer 非常擅长建立长程依赖，但近期研究通过直观的可视化解释表明，Transformer 缺乏捕捉高频成分的能力，而高频成分主要传递局部信息

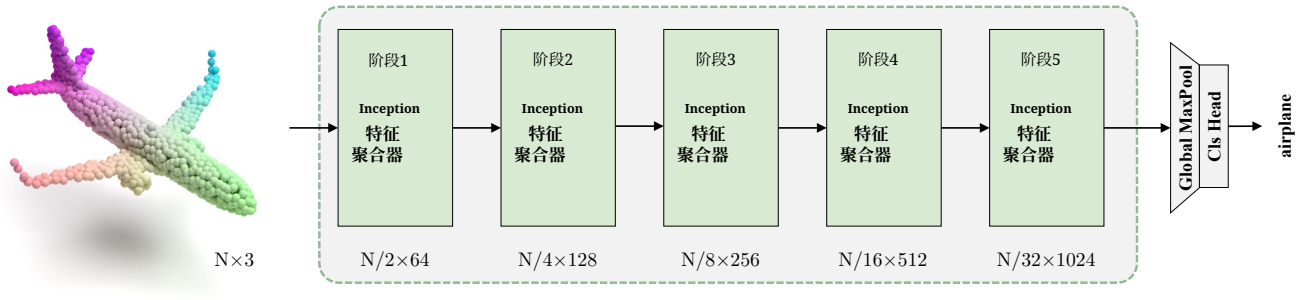


图2. PointCo的整体架构 ntent-based Transformer（PointConT）。该网络由一系列 Inception特征聚合器模块组成 .

[16], [17]。换言之，Transformer 是一种低通滤波器。为了解决这一问题，Inception Transformer（iFormer） [17]设计了一种通道拆分机制，采用并行卷积分支和自注意力分支，分别作为高频混合器和低频混合器。

受特征空间局部注意力和不同频率特征概念的启发，我们采用基于内容的Transformer和Inception特征聚合器进行三维点云分类。

III. 方法

A. 总体架构

所提出的 PointConT 架构概览如图 2所示。骨干结构由五个层级阶段的 Inception 特征聚合器模块组成。

给定输入点云 $p \in \mathbb{R}^{N\times 3}$，其中包含 N 个三维空间中的点。“阶段 1”的 Inception 特征聚合器模块将点云划分为重叠分块，然后将输入坐标嵌入新的特征空间（维度记为 C）。每个阶段的点数减半，特征维度加倍。因此，第 m 阶段的输出包含 $\frac{N}{2^m}$ 个点和 $2^{m-1}C$ 个特征维度。对于分类，最终的分类器头由全局最大池化和两个线性层组成。 —

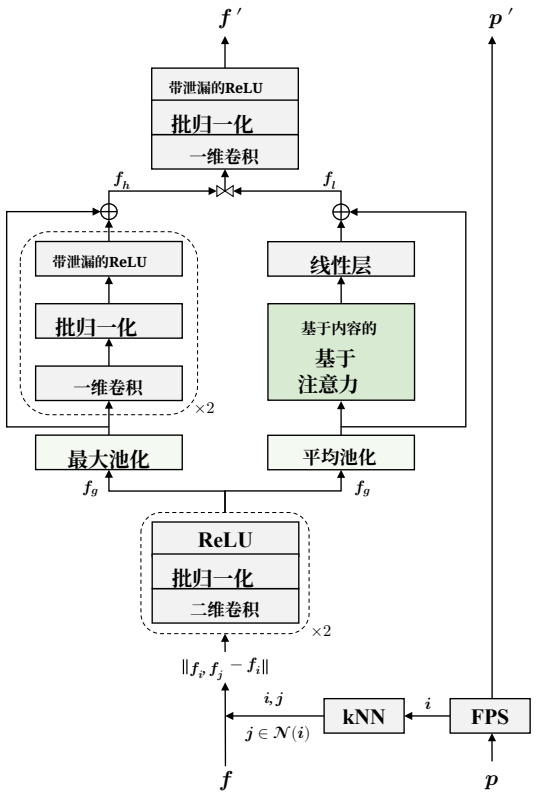


图 3. Inception特征聚合器的详细结构。

B. Inception 特征聚合器

如图 3所示，以第 m个$(m > 1)$ 阶段为例。给定来自上一阶段的点坐标 $p \in \mathbb{R}^{\frac{N}{2^{m-1}}\times 3}$ 和点特征 $f \in \mathbb{R}^{\frac{N}{2^{m-1}}\times 2^{m-2}C}$，Inception 特征聚合器模块首先以 2 倍的采样率从 p 通过最远点采样选取中心点，然后使用 k近邻算法对局部分块进行分组。我们将 i 视为中心点，并将$\{j : (i, j) \in \mathcal{N}\}$ 视为围绕该中心点的一个分块。我们进一步采用 DGCNN 中设计的 EdgeConv [37] （如式（1）所示），用于提取每个分块内中心点特征$f_i \in \mathbb{R}^{\frac{N}{2^m}\times 2^{m-2}C}$ 与其邻居点 $f_j \in \mathbb{R}^{\frac{N}{2^m}\times k\times 2^{m-2}C}$ —
(k表示邻居点的数量)之间的关系。

$$f_g = \text{MLP}(\|f_i, f_j - f_i\|), \quad f_g \in \mathbb{R}^{\frac{N}{2^m}\times k\times 2^{m-1}C} \tag{1}$$

其中 $f_j - f_i$ 表示将 f_j 减去 f_i，以获得相对于质心 i的邻域特征，$\|\cdot\|$是拼接操作，而多层感知机是一个简单的网络，包含一个逐点卷积层、一个批归一化层和一个激活函数。请注意，与在特征空间中定义 k近邻的 DGCNN 不同，我们采用在三维空间中的邻域搜索。

接下来，我们提出一种混合池化策略，用于聚合局部分块的特征。在大多数以往的研究中，最大池化已被证实能够有效聚合局部特征，因为它可以捕获主要传递局部信息的高频成分。PointConT 并未采用将最大池化与 Transformer块直接串联的方式，而是使用由高频聚合分支和低频聚合分支组成的并行结构。最大池化操作聚合高

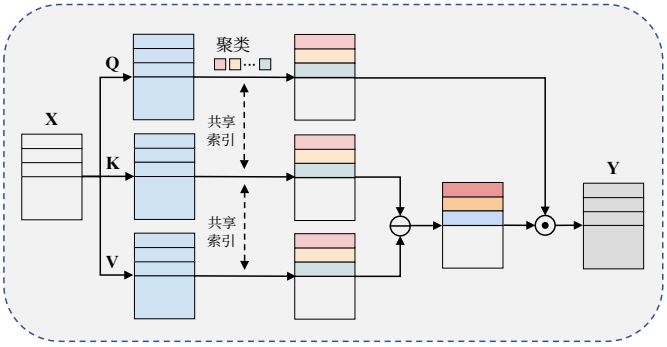


图 4. 基于内容的注意力示意图。它可以动态地将所有查询聚类为多个组，并在每个组内计算自注意力。

频率信号，而平均池化操作则滤除低频表示。

高频聚合分支：该分支可定义为

$$f_h = \text{ResMLP}(\text{MaxPool}(f_g)), \quad f_h \in \mathbb{R}^{\frac{N}{2^m} \times 2^{m-1}C} \tag{2}$$

其中，MaxPool 和 ResMLP 分别表示最大池化操作和残差多层感知机块。

低频聚合分支：我们只需在基于内容的 Transformer（ConT）之前使用一个平均池化层（AvgPool），这一设计使基于内容的 Transformer 能够专注于嵌入低频信息。该分支可定义为

$$f_l = \text{ConT}(\text{AvgPool}(f_g)), \quad f_l \in \mathbb{R}^{\frac{N}{2^m} \times 2^{m-1}C} \tag{3}$$

最后，我们将高频聚合分支和低频聚合分支的特征拼接起来，然后将其输入多层感知机块，作为 Inception 聚合器的输出特征 f'。

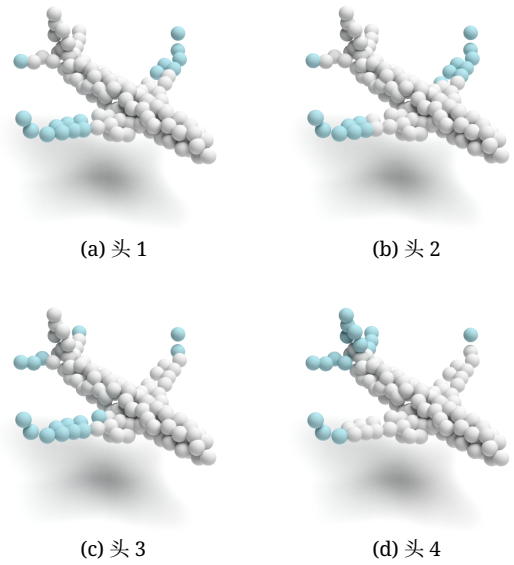
$$f' = \text{MLP}(\|f_h, f_l\|), \quad f' \in \mathbb{R}^{\frac{N}{2^m} \times 2^{m-1}C} \tag{4}$$

C. 基于内容的 *Transformer*

与 Point Transformer 不同， [8],后者在局部空间邻居点之间计算自注意力，而我们提出了一种基于内容的注意力机制，如图 4所示。该机制会在每个模块中根据查询的内容（即特征）动态地将所有查询划分为多个簇，并选择相应的键和值来计算局部自注意力。

设 $X \in \mathbb{R}^{S \times d}$（$d$ 为特征空间维度， S 表示特征的长度）是一组特征向量。此外，我们得到嵌入 $Q = XW_Q$、$K = XW_K$ 和 $V = XW_V$，分别表示查询、键和值。

然后，我们使用聚类算法，将查询分散到不同的簇中。K 均值聚类算法是解决聚类问题的经典方法。然而，K 均值聚类通常会使各簇包含数量不等的查询，因此无法通过图形处理器以并行方式实现该算法。为解决这一问题，我们参考了 BOAT [55], 提出的平衡二元聚类算法，该算法将一组查询分层均等地划分为两个簇。



(a) 头 1 (b) 头 2

(c) 头 3 (d) 头 4

图 5. 不同注意力头中的点聚类可视化。

通过聚类，将查询 Q 分组成 L个子集 $\{Q_i\}_{i=1}^{L}$，其中每个子集的大小相同 $|Q_i| =_L^S$。随后，按照相同的索引将键 K 和值 V 分别划分为 $\{K_i\}_{i=1}^{L}$ 和 $\{V_i\}_{i=1}^{L}$。每个子集中的自注意力（SA）表示为

$$Y_i = \text{SA}(Q_i, K_i, V_i) \tag{5}$$

其中 Y_i 是每个子集的输出。最后，所有子集$\{Y_i\}_{i=1}^{L}$ 按照原有顺序合并为输出 $Y \in \mathbb{R}^{S \times d}$。

多头配置是 Transformer 中的标准做法。我们扩展出多个注意力头，每个注意力头都独立执行查询/键/值嵌入和聚类。这种设置极大地提升了聚类的多样性，如图 5所示。

层次二元聚类：与 K-means 聚类类似，簇分配取决于所有簇质心与每个样本之间的距离；我们的二元聚类从将查询 $Q = \{q_i\}_{i=1}^{S}$随机划分为两个簇开始，然后计算这两个簇的质心，分别记为 c_1 和 c_2。之后，我们计算距离比以进行硬分配。上述操作可概括为

$$\begin{cases} c_1 = \dfrac{\sum\{q_i\}_{i=1}^{\frac{S}{2}}}{\frac{S}{2}}, \quad c_2 = \dfrac{\sum\{q_i\}_{i=\frac{S}{2}+1}^{S}}{\frac{S}{2}} \\ r_i = \dfrac{dist\,(q_i, c_1)}{dist\,(q_i, c_2)}, \forall i \in [1, S] \\ [i_1, \cdots, i_S] = \text{argsort}(\{r_i\}_{i=1}^{S}) \\ \mathcal{C}_1 = \{q_{i_j}\}_{j=1}^{\frac{S}{2}}, \quad \mathcal{C}_2 = \{q_{i_j}\}_{j=\frac{S}{2}+1}^{S} \end{cases} \tag{6}$$

其中， $dist$ 表示特征空间中的欧氏距离， $\mathcal{C}_1$和 $\mathcal{C}_2$ 表示通过平衡二元聚类得到的两个大小相同的簇。经过 n 次（$n = \log_2 L$）二元聚类后，我们得到 L 个大小相同的子集。

SA 的选择：在 Point Transformer 中，自注意力的选择对 Transformer 块的特性有着至关重要的影响。SA 的一种选择是标准的标量注意力，如下所示：

$$\mathrm{SA} = \mathrm{Softmax}(\frac{QK^T}{\sqrt{d}})V \tag{7}$$

SA 的另一种选择是向量注意力 [8], [56] ，本文采用了这种方法

$$\mathrm{SA} = \mathrm{Softmax}(\frac{Q-K}{\sqrt{d}}) \odot V \tag{8}$$

复杂度分析：给定一组大小为 $S \times d$ 的特征向量，对于标准多头自注意力（MSA），局部多头自注意力模块的计算复杂度为 $\Omega(S \times (4kd^2 + 2k^2d)) = 4Skd^2 + 2Sk^2d$，其中 k 是局部邻域中的点数。与标量注意力不同，向量注意力可将复杂度进一步降低至 $\Omega(4Skd^2 + 2Skd)$。在我们的 PointConT 中，分层二元聚类算法以不重叠的方式划分特征向量，从而将复杂度大幅降低至 $\Omega(4Sd^2 + 2Sd)$。

$$\begin{cases} \Omega(\mathrm{MSA_{Local}}) = 4Skd^2 + 2Sk^2d \\ \Omega(\mathrm{MSA_{PointTrans.}}) = 4Skd^2 + 2Skd \\ \Omega(\mathrm{MSA_{PointConT}}) = 4Sd^2 + 2Sd \end{cases} \tag{9}$$

IV. 实验

本节展示所提出模型在形状分类任务上的实验结果。所有实验均在一块 Tesla V100 GPU 上进行。

实现细节。我们使用 PyTorch 框架实现 PointConT，并采用 SGD优化器训练网络（动量和权重衰减分别设为 0.9 和 0.0001），余弦学习率调度的初始学习率设为 0.001（预热步数设为 10），并使用带标签平滑的交叉熵。所有实验均固定随机种子，以消除随机性的影响。

对于形状分类训练，我们仅使用 1024 个均匀采样的点作为网络输入。此外，除随机缩放和平移外，我们还使用 RSMix [57] 进行数据增强。我们在 ModelNet40 和 ScanObjectNN 上训练 PointConT，批量大小分别设为 32 和 64，训练轮数分别为 300 和 400。测试时，ModelNet40 和 ScanObjectNN 的批量大小分别设为 16 和 32。

A. *ModelNet40* 上的分类

我们在 ModelNet40 [20] 形状分类基准上评估该模型。该数据集包含来自 40 个常见类别的 12,308 个点云计算机辅助设计（CAD）模型，其中 9840 个模型用于训练，2468 个模型用于测试。

结果如表 I所示。PointConT 在 ModelNet40 上的总体准确率为 93.5%，在基于注意力的模型中表现出色。此外，我们的 PointConT 具有较高的推理速度（训练时每秒 166 个样本，测试时每秒 279 个样本），比原始 PointMLP 快 3。5×倍 [58]，比论文中的轻量版本快 1。4× 倍。我们在图 6中展示了各阶段的聚类结果。这些簇能够捕获长程依赖。

表 I ModelNet40 数据集上的形状分类结果。OA：总体准确率。

方法	模型	输入	总体准确率(%)
MLP	PointNet [27]	1K 个点+法向量	89.2
	PointNet++ [15]	1K 个点	90.7
	PointNet++ [15]	5K 个点+法向量	91.9
	PointMLP [58]	1K 个点	**94.1**
Conv	PointCNN [28]	1K 个点	92.5
	PointWeb [30]	1K 个点+法向量	92.3
	PointConv [29]	1K 个点+法向量	92.5
	RS-CNN [31]	1K 个点	92.9
	KPConv [34]	1K 个点	92.9
	PosPool [32]	5K 个点	93.2
	PAConv [35]	1K 个点	93.6
图	DGCNN [37]	1K 个点	92.9
	CurveNet [39]	1K 个点	93.8
非局部	PointASNL [44]	1K 个点	92.9
注意力	GDANet [40]	1000 个点	93.4
	PCT [9]	1000 个点	93.2
	点变换 [8]	1000 个点	93.7
	PointConT（我们的方法）	1K 个点	93.5

B. *ScanObjectNN* 上的分类

我们还在近期的真实世界点云分类数据集 ScanObjectNN[21], 上进行了实验，该数据集包含来自 15 个类别的 1.5 万个物体。我们仅报告刚性变换数据集 PB T50 RS 中重度扰动的数据。与 ModelNet40 中采样得到的虚拟点云不同，ScanObjectNN 中的物体来自真实世界扫描。因此，ScanObjectNN 中的点云含有噪声（背景、遮挡），且未与坐标轴对齐，这给现有点云分析方法带来了重大挑战。

表 II展示了 ScanObjectNN 上的分类结果。PointConT 优于先前的模型，在不使用投票的情况下总体准确率达到 88.0% [31]，并在对 10 次预测投票取平均后达到 90.3% 的 Top-1 准确率。这表明 PointConT 在真实世界点云中表现有效。

C. 消融研究

我们针对形状分类任务，对方法中的关键设计进行了消融研究。所有实验均在相同的训练设置下进行。

组件消融研究： 表III报告了移除 PointConT 中各个组件后的分类结果。比较实验III和实验IV，我们发现，采用基于内容的Transformer后，模型在 ModelNet40 上提升了 0.6%，在 ScanObjectNN 上提升了1.7%。这表明，基于内容的Transformer能够增强点云的表征能力。值得注意的是，实验V的结果大幅下降。在没有平均池化的情况下，实验V表示基于内容的Transformer以串行方式接在最大池化

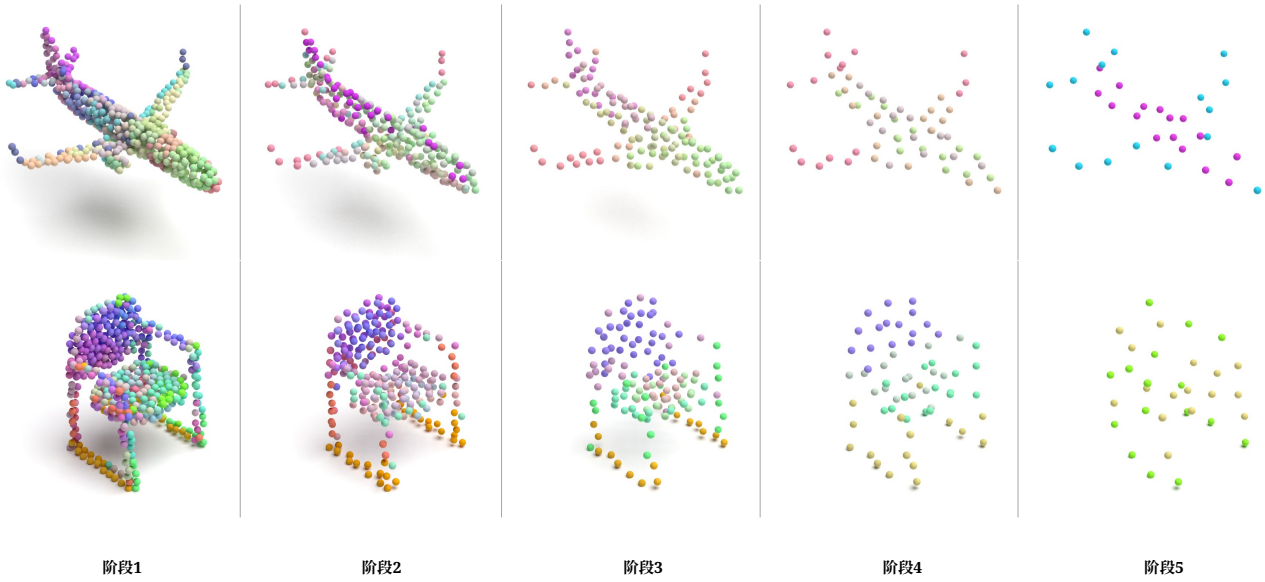


图 6. 各阶段聚类结果的可视化。同一簇中的点以相同颜色绘制，不同簇以随机颜色区分。

表 II ScanObjectNN 数据集上的形状分类结果。

*表示采用投票策略评估的方法 [31]。mAcc：平均类别准确率；OA：总体准确率。

模型	mAcc(%)	OA(%)
PointNet [27]	63.4	68.2
SpiderCNN [33]	69.8	73.7
PointNet++ [15]	75.4	77.9
PointCNN [28]	75.1	78.5
DGCNN [37]	73.6	78.1
GBNet [59]	77.8	80.5
SimpleView [26]	-	80.5
PRANet [60]	79.1	82.1
Point-BERT [61]	-	83.1
Point-MAE [62]	-	85.2
PointMLP [58]	84.4	85.7
PointNeXt [46]	86.8	88.2
PointConT（我们的方法）	86.0	88.0
PointConT（我们的方法）*	**88.5**	**90.3**

和残差多层感知机之后，这说明混合池化策略在 PointConT 中发挥着重要作用。显然，结合所有这些组件后，我们在 ModelNet40 和 ScanObjectNN 上取得了最佳结果，这表明基于内容的Transformer和Inception特征聚合器在点云分类中的有效性。

阶段数： 在 表 IV中，我们对 PointConT 中不同的阶段数进行了消融研究。我们逐步增加 PointConT 在 ModelNet40 和 ScanObjectNN 数据集上的深度，以检验增加深度的有效性。我们发现，5 个阶段已足以充分发挥模型的潜力。更深的模型会带来冗余信息并导致性能下降。

表III 消融研究。**最大池化**：最大池化；**残差多层感知机**：残差多层感知机；**平均池化**：平均池化；**基于内容的Transformer**：基于内容的Transformer。指标：总体准确率(%)。

ID	MP	Res	AP	ConT	ModelNet40	ScanObjectNN
I	✓				93.2	86.5
II	✓	✓			93.1	87.3
III	✓	✓	✓		92.9	86.3
IV	✓	✓	✓	✓	**93.5**	**88.0**
V	✓	✓		✓	92.8	87.2
VI	✓		✓	✓	92.8	87.2
VII			✓	✓	93.0	81.0

表 IV 消融研究：阶段数量。

阶段数量	ModelNet40	ScanObjectNN
3	91.9	84.9
4	92.7	86.8
5	**93.5**	**88.0**
6	92.8	86.3

局部簇大小： 我们探讨了局部簇大小的设置，并在表 V中展示结果。局部簇大小设为 16 时，性能最佳。

表 V 消融研究：局部簇大小。

局部簇大小	ModelNet40	ScanObjectNN
8	92.9	87.7
16	**93.5**	**88.0**
32	92.8	87.7

相似性度量：我们比较了用于聚类的两种重要相似性度量：余弦相似度和欧氏距离。余弦相似度与两个向量的点积成正比。因此，余弦相似度较高的向量从原点出发的方向较为接近；而欧氏距离对应于向量差的 L2 范数。欧氏距离较小的向量位于向量空间中较近的区域。 表 VI 中的结果表明，在分类任务中，根据欧氏距离进行聚类优于使用余弦相似度。

表 VI 消融研究：相似性度量。

相似性度量	ModelNet40	ScanObjectNN
余弦相似度	92.9	87.8
欧氏距离	**93.5**	**88.0**

注意力类型： 最后，我们比较了 第 III-C 节 中介绍的标量注意力和向量注意力。结果如表 VII所示。显然，注意力模块比无注意力模块更有效，而向量注意力略优于标量注意力。正如 Point Transformer 中所述，向量注意力支持对单个特征通道进行自适应调制，而非仅对整个特征向量进行调制，这有利于三维点云分析。

表 VII 消融研究：注意力类型。

注意力类型	ModelNet40	ScanObjectNN
无注意力	92.9	86.3
标量注意力	92.9	87.9
向量注意力	**93.5**	**88.0**

V. 结论

本文提出了点内容 Transformer（PointConT），这是一种简单而强大的架构，采用基于内容的Transformer，将具有相似特征的采样点聚类到同一类别中，并在每个类别内计算自注意力。与局部空间注意力相比，基于内容的Transformer能够建立长程特征依赖。此外，我们设计了一种 Inception特征聚合器，以并行方式结合高频和低频信息。最大池化操作聚合高频信号，而平均池化操作和 Transformer 滤波器则处理低频表示。我们希望本研究能为点云 Transformer 的设计提供有价值的见解。

值得注意的是，平衡聚类算法生成的簇大小相同，这限制了所提出的 PointConT 的通用性和灵活性。未来工作中可采用更先进的聚类方法和 CUDA 来实现逐簇矩阵乘法。

参考文献

[1] O. Schumann, J. Lombacher, M. Hahn, C. W¨ohler 和 J. Dickmann，《利用汽车雷达进行场景理解》，IEEE 智能车辆汇刊，第 5 卷，第 2 期，第 188–203 页，2020。[2] L. Li, M. Yang, L. Guo, C. Wang 和 B. Wang，《面向城市环境中智能车辆的基于分层邻域的精确定位》，IEEE 智能车辆汇刊，第 1 卷，第 3 期，第 220–229 页，2016。[3] X. S un, S. Shen, H. Cui, L. Hu 和 Z. Hu，《利用高分辨率倾斜航空影像对城市场景进行地理、几何和语义重建》，IEEE/CAA 自动化学报，第 6 卷，130，2019 1，118第。页 – 。[4] Z. Chen, J. Zhang 和 D. Tao，《用于道路检测的渐进式激光雷达适配》，IEEE/CAA 自动化学报，第 6 卷，702，2019 3，693 第。页 – 。[5] T.-H. Chen 和 T. S. Chang，《RangeSeg：面向三维激光雷达点云的范围感知实时分割》，IEEE 智能车辆汇刊，第 7 卷，第 1 期，第 93–101 页，2022。[6] C. Zhao, C. Fu, J. M. Dolan 和 J. Wang，《基于 L形拟合的车辆姿态估计与跟踪：使用三维激光雷达》，IEEE 智能车辆汇刊，第 6 卷，第 4 期，第 787–798 页，2021。[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser 和 I. Polosukhin，《Attention Is All You Need》，载于神经信息处理系统会议（NeurIPS）论文集，2017，第 5998–6008 页。[8] H. Zhao, L. Jiang, J. Jia, P. Torr 和 V. Koltun，《Point Transformer》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，12 2021，第 16 239–16 248 页。[9] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin 和 S.-M. Hu，《PCT：点云 Transformer》，计算视觉媒体，第 7 卷，第 187–199 页，2021。[10] X. Pan, Z. Xia, S. Song, L. E. Li 和 G. Huang，《利用 Pointformer 进行三维目标检测》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2021，第 7463–7472 页。[11] C. Park, Y. Jeong, M. Cho 和 J. Park，《Fast Point Transformer》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，12 2022，第 16 949–16 958 页。[12] X. Lai, J. Liu, L. Jiang, L. Wang, H. Zhao, S. Liu, X. Qi 和 J. Jia，《用于三维点云分割的分层 Transformer》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，3 2022，第 8500–8509 页。[13] C. Zhang, H. Wan, X. Shen 和 Z. Wu，《PVT：用于点云学习的点-体素 Transformer》，arXiv 预印本 arXiv:2108.06076，8 2021。[14] Z. Yang, L. Jiang, Y. Sun, B. Schiele 和 J. Jia，《点云理解的统一基于查询范式》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2022，第 8541–8551 页。[15] C. R. Qi, L. Yi, H. Su 和 L. J. Guibas，《PointNet++：度量空间中点集的深度分层特征学习》，载于神经信息处理系统会议（NeurIPS）论文集，6 2017，第 5100–5109 页。[16] N. Park 和 S. Kim，《视觉 Transformer 如何工作？》，载于国际学习表征会议（ICLR）论文集，2 2022。[17] C. Si, W. Yu, P. Zhou, Y. Zhou, X. Wang 和 S. Yan，《Inception Transformer》，载于神经信息处理系统会议（NeurIPS）论文集，2022。[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke 和 A. Rabinovich，《通过卷积不断深入》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，6 2015，第 1–9 页。[19] C. Szegedy, S. Ioffe, V. Vanhoucke 和 A. A. Alemi，《Inception-v4、Inception-ResNet 及残差连接对学习的影响》，载于 AAAI 人工智能会议（AAAI）论文集，2017，第 4278–4284 页。[20] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang 和 J. Xiao，《3D ShapeNets：体积形状的深度表示》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2015，第 1912–1920 页。[21] M. A. Uy, Q.-H. Pham, B.-S. Hua, D. T. Nguyen 和 S.-K. Yeung，《重新审视点云分类：真实世界数据上的新基准数据集和分类模型》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，2019，第 1588–1597 页。[22] Y. Zhou 和 O. Tuzel，《VoxelNet：用于基于点云的三维目标检测的端到端学习》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2018，第 4490–4499 页。[23] D. Maturana 和 S. Scherer，《VoxNet：用于实时目标识别的三维卷积神经网络》，载于 IEEE/RSJ 智能机器人与系统国际会议（IROS）论文集，2015，第 922–928 页。

[24] G. Riegler, A. O. Ulusoy 和 A. Geiger, 《OctNet：高分辨率下深度三维表示的学习》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集, 2017，第 6620–6629 页。[25] H. Su, S. Maji, E. Kalogerakis 和 E. Learned-Miller, 《用于三维形状识别的多视图卷积神经网络》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，2015，第 945–953 页。[26] A. Goyal, H. Law, B. Liu, A. Newell 和 J. Deng, 《重新审视点云形状分类：一种简单有效的基线方法》，载于国际机器学习会议（ICML）论文集, 2021，第 3809–3820 页。[27] C. R. Qi, H. Su, K. Mo 和 L. J. Guibas, 《PointNet：点集上的深度学习，用于三维分类与分割》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集。电气与电子工程师学会，7 2017，第 652–660 页。[28] Y. Li, R. Bu, M. Sun, W. Wu, X. Di 和 B. Chen, 《PointCNN：对 X变换点进行卷积》，载于神经信息处理系统会议（NeurIPS）论文集，S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi 和 R. Garnett 编，第 31 卷。Curran Associates, Inc., 2018，第 828–838 页。[29] W. Wu, Z. Qi 和 L. Fuxin, 《PointConv：三维点云上的深度卷积网络》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2019，第 9621–9630 页。[30] H. Zhao, L. Jiang, C.-W. Fu 和 J. Jia, 《PointWeb：增强点云处理的局部邻域特征》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2019，第 5565–5573 页。[31] Y. Liu, B. Fan, S. Xiang 和 C. Pan, 《用于点云分析的关系形状卷积神经网络》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2019，第 8895–8904 页。[32] Z. Liu, H. Hu, Y. Cao, Z. Zhang 和 X. Tong, 《深入考察点云分析中的局部聚合算子》，载于欧洲计算机视觉会议（ECCV）论文集，7 2020，第 326–342 页。[33] Y. Xu, T. Fan, M. Xu, L. Zeng 和 Y. Qiao, 《SpiderCNN：使用参数化卷积滤波器对点集进行深度学习》，载于欧洲计算机视觉会议（ECCV）论文集，2018，第 90–105 页。[34] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette 和 L. J. Guibas, 《KPConv：面向点云的灵活可变形卷积》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，4 2019，第 6410–6419 页。[35] M. Xu, R. Ding, H. Zhao 和 X. Qi, 《PAConv：点云上的位置自适应卷积与动态核组装》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2021，第 3172–3181 页。[36] J. Li, B. M. Chen 和 G. H. Lee, 《SO-Net：用于点云分析的自组织网络》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2018 年 6 月，第 9397–9406 页。[37] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein 和 J. M. Solomon, 《用于点云学习的动态图卷积神经网络》，ACM 图形学汇刊，第 38 卷，第 1–12 页，10 2019。[38] C. Wang, B. Samari 和 K. Siddiqi, 《用于点集特征学习的局部谱图卷积》，载于欧洲计算机视觉会议（ECCV）论文集，9 2018，第 56–71 页。[39] T. Xiang, C. Zhang, Y. Song, J. Yu 和 W. Cai, 《漫步云中：用于点云形状分析的曲线学习》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，5 2021，第 895–904 页。[40] M. Xu, J. Zhang, Z. Zhou, M. Xu, X. Qi 和 Y. Qiao, 《学习几何解耦表示，以全面理解三维物体点云》，载于 AAAI 人工智能会议（AAAI）论文集，2021，第 3056–3064 页。[41] H. Ran, W. Zhuo, J. Liu 和 L. Lu, 《学习点云上的组内关系》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，2021，第 15 457–15 467 页。[42] S. Fan, Q. Dong, F. Zhu, Y. Lv, P. Ye 和 F.-Y. Wang, 《SCF-Net：学习空间上下文特征以进行大规模点云分割》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2021，第 14 499–14 508 页。[43] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin 和 B. Guo, 《Swin Transformer：使用移位窗口的分层视觉 Transformer》，载于 IEEE 国际计算机视觉会议（ICCV）论文集，2021，第 10 012–10 022 页。[44] X. Yan, C. Zheng, Z. Li, S. Wang 和 S. Cui, 《PointASNL：利用自适应采样和非局部神经网络实现稳健点云处理》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，3 2020，第 5588–5597 页。[45] X. Wang, R. Girshick, A. Gupta 和 K. He, 《非局部神经网络》，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2018，第 7794–7803 页。

[46] G. Qian、Y. Li、H. Peng、J. Mai、H. A. A. K. Hammoud、M. Elhoseiny 和 B. Ghanem, “PointNeXt：重新审视 PointNet++，改进训练与扩展策略”，载于神经信息处理系统会议论文集（*NeurIPS*），2022年6月。[47] A. Dosovitskiy、L. Beyer、A. Kolesnikov、D. Weissenborn、X. Zhai、T. Unterthiner、M. Dehghani、M. Minderer、G. Heigold、S. Gelly、J. Uszkoreit 和 N. Houlsby, “一张图像胜过16×16个词：用于大规模图像识别的Transformer”，载于国际学习表征会议论文集（*ICLR*），2021年。[48] W. Wang、E. Xie、X. Li、D.-P. Fan、K. Song、D. Liang、T. Lu、P. Luo 和 L. Shao, “金字塔视觉Transformer：一种无需卷积的通用密集预测骨干网络”，载于*IEEE*国际计算机视觉会议论文集（*ICCV*），2021年，第568–578页。[49] W. Wang、E. Xie、X. Li、D.-P. Fan、K. Song、D. Liang、T. Lu、P. Luo 和 L. Shao, “PVTv2：基于金字塔视觉Transformer的改进基线”，计算视觉媒体，第8卷，第3期，第1–10页，2022年。[50] K. Han、A. Xiao、E. Wu、J. Guo、C. Xu 和 Y. Wang, “Transformer中的Transformer”，载于神经信息处理系统会议论文集（*NeurIPS*），2021年2月，第15 908–15 919页。[51] X. Dong、J. Bao、D. Chen、W. Zhang、N. Yu、L. Yuan、D. Chen 和 B. Guo, “CSWin Transformer：一种采用十字形窗口的通用视觉Transformer骨干网络”，载于*IEEE*计算机视觉与模式识别会议论文集（*CVPR*），2022年，第12 124–12 134页。[52] A. Hassani、S. Walton、J. Li、S. Li 和 H. Shi, “邻域注意力Transformer”，*arXiv*预印本 *arXiv:2204.07143*，2022年。[53] Z. Huang、Y. Ben、G. Luo、P. Cheng、G. Yu 和 B. Fu, “Shuffle Transformer：重新思考视觉Transformer中的空间重排”，*arXiv*预印本 *arXiv:2106.03650*，2021年。[54] K. Liu、T. Wu、C. Liu 和 G. Guo, “动态组Transformer：一种具有动态组注意力的通用视觉Transformer骨干网络”，载于国际人工智能联合会议论文集（*IJCAI*），2022年3月，第1187–1193页。[55] T. Yu、G. Zhao、P. Li 和 Y. Yu, “BOAT：双边局部注意力视觉Transformer”，*arXiv*预印本 *arXiv:2201.13027*，2022年。[56] H. Zhao、J. Jia 和 V. Koltun, “探索图像识别中的自注意力”，载于*IEEE*计算机视觉与模式识别会议论文集（*CVPR*），2020年，第10 073–10 082页。[57] D. Lee、J. Lee、J. Lee、H. Lee、M. Lee、S. Woo 和 S. Lee, “通过刚性混合样本对点云进行正则化的策略”，载于*IEEE*计算机视觉与模式识别会议论文集（*CVPR*）。IEEE计算机学会，2021年2月，第15 895–15 904页。[58] X. Ma、C. Qin、H. You、H. Ran 和 Y. Fu, “重新思考点云中的网络设计与局部几何：一种简单的残差多层感知机框架”，载于国际学习表征会议论文集（*ICLR*），2022年2月。[59] S. Qiu、S. Anwar 和 N. Barnes, “用于点云分类的几何反投影网络”，*IEEE*多媒体汇刊，第24卷，第1943–1955页，2022年。[60] S. Cheng、X. Chen、X. He、Z. Liu 和 X. Bai, “PRA-Net：用于三维点云分析的点关系感知网络”，*IEEE*图像处理汇刊，第30卷，第4436–4448页，2021年。[61] X. Yu、L. Tang、Y. Rao、T. Huang、J. Zhou 和 J. Lu, “Point-BERT：通过掩码点建模预训练三维点云Transformer”，载于*IEEE*计算机视觉与模式识别会议论文集（*CVPR*），2022年11月，第19 313–19 322页。[62] Y. Pang、W. Wang、F. E. H. Tay、W. Liu、Y. Tian 和 L. Yuan, “用于点云自监督学习的掩码自编码器”，载于欧洲计算机视觉会议论文集（*ECCV*），2022年。