



上海科技大学
ShanghaiTech University

本科毕业论文（设计）

题 目： 基于大规模视觉语言预训练模型
的三维点云识别

学生姓名： 冯宇凡

学 号： 2019533141

入学年份： 2019

所在学院： 信息科学与技术学院

攻读专业： 计算机科学与技术

指导教师： 杨思蓓

上海科技大学

2023 年 04 月



上海科技大学
ShanghaiTech University

THESIS

Subject: 3D Point Cloud Understanding based on Large Vision-Language Pretrained Models

Student Name: Yufan Feng

Student ID : 2019533141

Year of Entrance: 2019

School: School of Information Science and Technology

Major: Computer Science and Technology

Advisor: Sibei Yang

ShanghaiTech University

Date: 04 / 2023

上海科技大学

毕业论文(设计)学术诚信声明

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者签名：汤宇凡

日期：2023年4月21日

上海科技大学

毕业论文（设计）版权使用授权书

本毕业论文（设计）作者同意学校保留并向国家有关部门或机构递交论文的复印件和电子版，允许论文被查阅和借阅。本人授权上海科技大学可以将本毕业论文（设计）的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本毕业论文（设计）。

保 密，在__年解密后适用本授权书。

本论文属于

不保密。

（请在以上方框内打“√”）

作者签名：汤序凡

指导教师签名：林海平

日期：2023 年 4 月 21 日

日期：2023 年 4 月 21 日

摘要

随着深度学习技术的快速发展，大规模预训练模型在自然语言和视觉领域取得了显著成果，例如 BERT、GPT 和 ViT 等。其中 CLIP 模型通过将图像和文本信息映射到同一向量空间，实现跨模态的语义匹配。然而，在三维视觉领域，由于数据稀缺，应用这种预训练大型模型和微调下游任务的范式遇到了挑战。因此，本研究旨在探讨利用视觉-语言预训练模型，如 CLIP，来增强对三维数据理解的能力。为实现这一目标，我们需要克服点云数据与图像数据之间的巨大差距，例如图片带有丰富的结构、边线和颜色信息，而点云则是稀疏、无序的集合。因此，如何将三维数据的输入转变为可被 CLIP 识别的视觉特征，并充分利用 CLIP 中的视觉信息成为了主要的挑战。尽管早期的 PointCLIP 工作将点云投影到一系列深度图并输入到 CLIP 的二维视觉编码器中，但它并不能充分地利用三维数据中的几何信息，因此并不能作为一个基础模型来使用。本工作的贡献在于提出一个崭新的模型框架，以克服这些挑战，通过将三维数据输入转换为可被 CLIP 识别的视觉特征，我们成功地将 CLIP 模型应用于三维点云数据，从而在下游任务上取得了显著的性能提升。同时我们探究了如何将模型迁移到点云理解问题上的同时，保持 CLIP 泛化性能。我们的研究结果表明，所提出的模型框架在三维视觉任务中表现出了优异的性能。此外，我们还证明了所提出的方法在少样本和零样本训练条件下仍具有的泛化能力，这在实际应用中具有重要的价值。

关键词： 视觉-语言预训练模型，三维点云理解，迁移学习

Abstract

As deep learning technology advances rapidly, large-scale pre-trained models such as BERT, GPT, and ViT, have achieved remarkable results in the natural language and visual domains. The CLIP model maps images and text information to the same vector space, realizing cross-modal semantic matching. However, in the 3D vision field, the paradigm of applying this paradigm of pre-training and fine-tuning is challenging due to data scarcity. Thus, this study aims to explore the use of visual-language pre-trained models, such as CLIP, to enhance the ability of understanding 3D data. To achieve this goal, we need to overcome the huge gap between point cloud data and image data, such as images with rich structural, boundary, and color information, while point clouds are sparse and unordered sets. Therefore, how to transform the input of 3D data into visual features that can be recognized by CLIP and fully utilize the visual information in CLIP becomes the main challenge. Although early PointCLIP work projected point clouds into a series of depth maps and input them into CLIP’s 2D visual encoder, it did not fully utilize the geometric information in 3D data, so it cannot be used as a basic model. The contribution of this work is to propose a new model framework to overcome these challenges. By transforming the input of 3D data into visual features that can be recognized by CLIP, we successfully applied the CLIP model to 3D point cloud data, achieving significant performance improvements in downstream tasks. At the same time, we explored how to transfer the model to point cloud understanding problems while maintaining the generalization performance of CLIP. Our research results demonstrate that the proposed model framework exhibits excellent performance in 3D vision tasks. In addition, we also demonstrate that the proposed method still has generalization ability under few-shot and zero-shot training schemes, which has important value in practical applications.

Keywords: Vision-language pre-trained models, point cloud understanding, transfer learning

目 录

第一章 引言	1
第二章 相关工作	3
2.1 对点云表示的学习	3
2.1.1 基于点的方法	3
2.1.2 基于体素的方法	4
2.1.3 基于投射图的方法	4
2.2 三维点云上的预训练模型	5
2.3 CLIP 和其相关工作	6
第三章 实验方法	8
3.1 在语义层面的特征对齐	8
3.2 在输入层面的特征对齐	10
3.2.1 投射图像和渲染图像的对应问题	11
3.2.2 构造图像块	13
3.2.3 对比学习的策略	15
第四章 实验结果	16
4.1 在语义层面的特征对齐的实验结果	16
4.2 在输入层面的特征对齐的实验结果	17
第五章 结论	19
参考文献	20
致谢	23

第一章 引言

随着深度学习技术的不断发展，神经网络的模型大小、特征表达的复杂程度都在逐渐升高。在这个背景下，大规模的预训练模型应运而生，并逐渐成为了一大研究热点。这些模型可以先在大规模的数据集上进行预训练，学习到丰富的特征表达和语义信息；之后再在一些具体的任务上用相对较小量级的数据集、甚至在少样本或者零样本数据上进行微调训练，从而在下游任务中获得优异的表现。例如，在自然语言领域，涌现了 BERT^[1]、GPT^[2]、T5^[3] 等基础模型，它们能够在自然语言理解、机器翻译、问答系统等多种文本任务中取得很好的表现；而视觉领域中，也有以 ViT^[4] 为代表的一系列纯视觉模型，他们在图像分类、目标检测和分割等各种下游任务上也展现出了优秀的能力。

CLIP 模型^[5] 则不同于之前的单一模态模型。它在超过四亿的图像-文本对的数据集上进行预训练，将图片和文本的信息映射到同一个向量空间，从而实现跨模态的语义匹配，具有了感知高质量表示的能力和一定的可迁移性。训练数据量的庞大，使得 CLIP 模型本身可以应对各种开放词汇的任务；而图像和语言域之间的固有对齐使得它能被广泛地应用于各种二维图像的下游任务中，如图像分类、图像分割、图片问答等，并且都在零样本、小样本等不同的训练策略下取得了优异的性能。

三维的视觉理解在生活中的各种新兴的研究领域都有很多的应用，如机器人技术、增强现实和自动驾驶等。理解三维空间中点的语义和特征对于解决下游理解的任务是至关重要的。然而，不同于上述的自然语言处理方向和二维视觉方向，由于难于获取丰富的三维的数据，在三维视觉领域中应用这种预训练大型模型和微调下游任务的范式遇到了障碍。之前也有一些工作应用了预训练-微调的范式，如 Point Contrast^[6]、Point-BERT^[7]、Point-MLP^[8] 等，它们证明了这种范式在三维领域也可以达到优秀的效果。然而，这些工作都是在数据集 ShapeNet^[9] 进行的预训练，该数据集中仅包含五万个人工合成的物体渲染模型，使得模型的可迁移性受到了限制，并会导致优化不足或者过拟合的问题。

在这样的背景下，我们不由得好奇：我们是否可以利用例如 CLIP 的视觉-语言预训练模型，来增强我们对三维数据的理解呢？如果我们可以，那我们就获得了一个可以迁移到三维空间的基础模型，它不但有助于提高现有的三维骨干网络的性能，还可以实现开放的三维场景的理解。然而，点云数据与图像数据之间

存在着巨大的间隔：图片带有丰富结构、边线、颜色等信息，而点云则是稀疏、无序的集合。于是，如何将三维数据的输入转变为可被 CLIP 识别的视觉特征、并充分利用 CLIP 中的视觉信息就成为了主要的难题。最早的将 CLIP 迁移至点云理解任务的工作是 PointCLIP^[10]，它将点云直接投影到了一系列的深度图并输入到 CLIP 的二维视觉编码器中。虽然它提供了一种直接而有效的解决方案，然而它并不能充分地利用三维数据中的几何信息，因此并不能作为一个基础模型来使用。

因此，本工作的意义在于提出一个崭新的模型框架，在保持 CLIP 泛化性能的同时，利用 CLIP 模型完成在三维点云上的下游任务。其中一个直观的任务就是在 ModelNet40 的分类数据集上，使用部分类的少部分数据，使得模型具有分类的能力。

第二章 相关工作

2.1 对点云表示的学习

当涉及到三维空间的分析和处理时，点云是一种常见的表达形式，它是三维空间中的一组点，或者说是一系列坐标、颜色的集合。点云具有无序性，即点云中的点没有任何特定的顺序；具有几何不变性，点云的表示与点云所在的位置、旋转和缩放无关；点云是不规则的，它不遵循任何特定的拓扑结构，因此需要使用一些特殊的方法来处理它们。这些特性使得点云难以进行分析和处理，因此如何对点云进行有效的特征提取和表示学习是点云分析领域一个重要的研究方向。近年来，基于深度学习的表示学习方法主要分为三种思路：基于点的方法、基于体素的方法和基于投射图的方法。

2.1.1 基于点的方法

在基于点的方法中，PointNet^[11]首次提出了一种在点云上直接进行端到端学习的方法，避免了繁琐的手工特征提取工作。PointNet采用多层次感知器（MLP）网络来处理点云数据，并使用全局最大池化来聚合每个点的特征，从而得到整个点云的表示。此外，PointNet还采用了对称函数来处理点的排列无关性，从而使点云对输入点云的旋转、平移和缩放不变。它是一种简单而有效的方法，同时具有较强的泛化能力，可以用于各种形状和大小的点云数据集。PointNet++^[12]是在PointNet基础上的改进工作，它在PointNet的基础上提出了一种新的方法来处理不同尺度和层次的点云数据，能够更好地捕捉点云数据地局部特征。PointNet++的架构主要包括两个主要的模块：PointNet集合抽象（PNSA）和PointNet特征回传（PNFP）。前者负责对点云数据多层次地聚合同步信息，而后者则负责将聚合后的局部特征传回原始的点云数据中，因此使得PointNet++能够更好地捕获局部特征，从而在点云分类和分割等任务中取得更好的效果。DGCNN^[13]也是一种基于PointNet的改进，它也使用了类似的局部领域聚合和全局池化的思路，并且使用了类似的初始特征提取网络，只是在特征提取、聚合方式方式上有所不同。在特征提取方面，DGCNN引入了k邻近图（k-NN）和图卷积神经网络（GCN），对每个点的邻域进行k近邻搜索，然后构建基于k近邻的无向图。在这个图上使用图卷积来聚合邻域特征，并在不同层级上重复这个过程。在特征提取方面，DGCNN在每个图卷积层中对每个节点的特征进行线性变换和非线性变化，使用图卷积更新每个节点的特征。

2.1.2 基于体素的方法

基于体素的一系列方法则会先将点云转换为体素网格的数据，将三维空间离散化，分成一个个小立方单元（体素），然后使用三维的卷积神经网络来对每个体素进行特征提取。这样，点云变成了类似于图片的密集数据，可以借鉴很多图片上的网络架构和方法。其中，最为基本的框架为 NVIDIA 提出的 Minkowski Engine^[14]。它是一个用于高效处理稀疏张量的 PyTorch^[15] 深度学习框架，能够有效地处理稀疏的点云数据。它主要基于 Minkowski 空间的思想，将 3D 空间划分为等距的体素，然后通过使用空间变换器网络（STN）来对输入的稀疏张量进行坐标变换和采样，从而在保持输入数据稀疏性的同时，使得输入数据可以在密集网格上进行处理。这种表示方法不仅可以节省内存和计算资源，还能够有效地处理稀疏的点云数据和体数据。MinkowskiNet^[14] 是基于 Minkowski Engine 的一个具体实现，它在 Minkowski Engine 的基础上提出了一种新的神经网络架构，用于处理 3D 点云数据和体数据。MinkowskiNet 的网络架构主要包括两个主要的模块：MinkowskiNet 特征提取和分类。前者采用多个 3D 卷积层和 3D 池化层来提取特征，其中每个卷积层和池化层都是基于 Minkowski Engine 实现的。后者采用全局最大池化和多层感知器（MLP）网络来对提取的特征进行分类。MinkowskiNet 的架构具有较强的可扩展性和通用性，可以用于处理各种大小和形状的 3D 数据，同时还能够实现较高的计算效率和模型准确性。PointConv^[16] 是在 Minkowski Engine 基础上实现的一个代表性的工作。它使用一组规则的、不重叠的球体作为体素来对点云进行划分，然后在每个体素上应用卷积运算，以提取局部特征。与传统的 3D 卷积不同，PointConv 可以根据点的分布情况自适应地调整卷积核的大小和形状，从而更好地适应不同形状和密度的点云数据。此外，PointConv 还提出了一种自适应的、距离加权的平均池化方法，用于将局部特征聚合成全局特征。PointConv 在各种点云数据集上的实验结果表明，它具有很强的特征学习能力和泛化能力，并且可以处理大规模的点云数据。

2.1.3 基于投射图的方法

另外，将点云投影到不同的二维图像平面上、转换为投影图也是一种解决点云稀疏性、无序性的方法。这种方法的优点在于可以直接使用现有的图像处理技术和比较成熟的二维卷积神经网络模型来处理点云数据，同时也提高了点云数据的处理效率。比如，MVCNN^[17] 将三维形状转换为多个渲染的视图。它由两个卷积神经网络（CNN）组成，第一个网络用于从每个视角的图片中提取试图特征，

第二个网络用于从池化后的所有试图特征中得到最终的物体表示。GVCNN^[18]是对 MVCNN 的改进，它在每个是叫上使用三维卷积神经网络来提取特征，并使用自适应的全局池化来聚合视图特征，更好地处理了三维物体中的局部细节。SimpleView^[19]则是一种简单而有效的方法，它将点云中的每个点投影到三个平面（XY、YZ 和 XZ 平面）上，将这三个平面的投影图作为输入，利用二维领域经典的 ResNet 系列架构得到每个视图的特征，最终在准确率和速度上都有很好的表现。与之前固定视角的多视角视图方法不同，MVTN^[20]运用可微分的渲染算法，提出了一种可以学习使用最优的视角的方法。

总的来说，基于点的方法、基于体素的方法和基于投影图的方法各自具有不同的优缺点和适用范围。基于点的方法适用于处理点云数据，能够有效地处理不规则和稀疏的点云数据，但对于处理大型点云数据的计算复杂度较高。基于体素的方法适用于处理 3D 体数据，能够节省计算资源和内存空间，但对于处理不规则和稀疏的点云数据的效果较差。基于投影图的方法能够利用二维卷积操作处理数据，但如何使得二维图像的基础网络有效识别三维数据是一大难点。

2.2 三维点云上的预训练模型

受二维图像上预训练模型的启发，研究者们也做了很多在三维数据上预训练的尝试。很多工作基于了自编码器（auto-encoder）将点云特征在预训练的过程中重建回原来的点云，从而获得对点云信息的理解。比如 OcCo^[21] 提出了一种无监督的点云预训练的方法。它构建了一个编码器-解码器的结构，在预训练过程中学习补全由不同相机视角下被遮挡的点云，之后使用预训练好的编码器权重作为下游点云任务微调的初始化。Point-BERT^[7] 将 BERT^[1] 的概念推广到点云。具体来说，它将点云划分为若干个局部的点集，使用一个离散变分自编码器（dVAE）将这些点集转化为包含有局部信息的词例（token）。之后，与 BERT 类似，Point-BERT 将这些词例的随机遮挡后，作为点 Transformer 的输入，在预训练中恢复出原始的点词例。在此基础上，进一步出现了 PointMAE^[22]，将 Point-BERT 训练起来较为繁琐的 dVAE 部分替换为由 PointNet 层组成的点词例嵌入层，并优化了自监督训练的策略。之后，Point-M2AE^[23] 将编码器和解码器修改为了金字塔结构，多尺度地重建点云的全局和局部，获得了更好的点云表达。

另一些工作使用了对比学习的策略。如 Point Contrast^[6] 将一个点云划分为多个局部的区域，对于每个区域通过不同的视角进行数据增强。之后，该方法使用一个编码器网络来提取局部区域的特征，之后在编码后的隐空间上使用对比

损失，使得同一个局部区域的特征互相更靠近。也有一些工作注意到了其他模态上预训练模型所包含的丰富的信息。如 CrossPoint^[24]、Simipu^[25]、MVI^[26] 借用了其他模态如二维图像、三维网格的现有神经网络，使用自监督的对比学习，使得同一个物体不同模态的表达互相靠近的同时，区分不同物体的表达。

最近，在文章 Can We Solve 3D Vision Tasks Starting from A 2D Vision Transformer?^[27] 中，作者提出了 Simple3D-Former，将二维的 ViT 图像嵌入层扩展成三维体素嵌入层，将二维的位置编码器替换成三维的位置编码器。证明了利用二维的 Transformer 解决三维理解任务的可行性。而 P2P^[28] 将点云投射为投影图之后，加入了一个可训练的上色模块，并将投影图输入到预训练好的图像模型中，在各种全监督的任务上取得了不错的表现。这些方法都尝试使用预训练好的二维的骨干网络来直接解决三维的问题，证明了后续工作的可行性。

2.3 CLIP 和其相关工作

CLIP^[5] 是 2021 年 1 月由 OpenAI 提出的一个多模态模型。它由一个图片编码器和一个文字编码器组成，图片编码器使用了基于 ResNet^[29] 的卷积神经网络和基于 Vision Transformer^[4] 的自注意力网络，而文字编码器使用了一个基于 Transformer^[30] 的自注意力网络。这两个编码器分别将图片和文字编码到一个共同的向量空间，在训练过程中，使得匹配的文本-图像对之间的余弦相似度最大化，于是可以使得匹配的文字和图片在向量空间中互相靠近。通过在高质量的包含由四亿对图像-文本对的数据集上进行训练，该模型可以将描述相似场景的图片和文字联系起来，从而可以零样本、或少样本地直接用于很多任务。例如，如果要进行零样本的 k 类图像分类，我们可以构造 k 个形如“<类名>的图像”或“<类名>的一张照片”的文本提示，将其用文本编码器编码获得 $Z_T \in \mathbb{R}^{k \times d}$ (其中 d 为特征向量的维度)；同时将需要分类的图像输入到图像编码器中获得 $Z_I \in \mathbb{R}^{1 \times d}$ 。接下来，我们可以直接利用点积 $\text{Logits} = Z_I Z_T^T$ 计算两者之间的相似度，之后利用 softmax 函数得到一个概率分布，选取概率最高的图片作为结果。

CLIP 的可迁移性使得它可以应用在很多任务上。在图像生成方面，出现了 DALL-E 2^[31]、Stable Diffusion^[32] 等，他们使用了 CLIP 训练好的与图像对齐的文本编码器，给扩散模型以文本的引导，使得其生成符合给定文本的图像。VideoCLIP^[33]、ActionCLIP^[34] 等工作将 CLIP 扩展到了 Video 上，AudioCLIP^[35] 和 WAV2CLIP^[36] 则将其拓展到了声音上。在三维领域，PointCLIP^[10] 首先将 CLIP 用于了点云的理解。它借鉴了 SimpleView^[19] 的方法，将每个物体的点云分别投

影到三个平面上，输入给 CLIP。PointCLIP 证明了 CLIP 对于点云的投射图的零样本分类任务也有一定的能力。此外，对于少样本任务，PointCLIP 还引入了一个由多层感知器 (MLP) 构成的适配器 (Adapter)，将其接在 CLIP 的图片编码器后续，可以使得点云投射图的向量域和 CLIP 原本的文本域对齐。PointCLIP 在点云的少样本学习的 ModelNet40 分类等任务上也达到了令人惊叹的效果。

第三章 实验方法

为了直观地测试模型对点云的理解，我们以开放词汇的点云识别问题为例。具体来说，我们将对一个点云的标注 \mathcal{Y} 分为基类 C^B 和新类 C^N 。在训练阶段，模型只会在标注为 \mathcal{Y}^B 的物体点云 $\mathcal{P} \in \mathbb{R}^{N \times 3}$ 上训练；而在测试阶段，模型则需要在所有的物体点云上分类。另外，我们还引入了一个较为普遍的少样本学习问题设置，即在训练集中，我们仅引入每一个类的几个样本。

3.1 在语义层面的特征对齐

PointCLIP^[10] 中提出了一个适配器的结构。具体来说，PointCLIP 将点云 \mathcal{P} 分别投影到 M 个投影图得到 $I_1, I_2, \dots, I_M \in \mathbb{R}^{H \times W \times C}$ ，将这些投影图直接输入到 CLIP 预训练好的图像编码器中，获得每个视角图的特征向量 $Z_1, Z_2, \dots, Z_M \in \mathbb{R}^{1 \times d}$ 。之后，PointCLIP 引入了一个适配器的结构，使得每个视角的特征融入其他视角的全局信息，通过这种方式获得了新的特征向量 $Z'_1, Z'_2, \dots, Z'_M \in \mathbb{R}^{1 \times d}$ ，并直接用这些向量与文本向量匹配相似度。训练时，PointCLIP 将 CLIP 的文字编码器和视觉编码器的权重都冻结住，只在适配器的权重上做调整，在最后的分类结果上加入交叉熵损失，监督分类结果的正确性。

然而，CLIP 对投影图的零样本分类的准确率并不高，即 CLIP 的编码器并不能很好地“认识”这些投影的图片，因此使用分类的交叉熵损失重复地训练简单的适配器结构，会使得适配器本身变成一个对于图片特征的分类器，导致使用开放词汇的点云识别设置时，已经训练好的适配器无法识别新类的物体。基于此，一个直观的解决方案是引入较为接近的但更为真实的同类别图片，使得图像特征之间相互靠近，而非直接与文字的标签信息匹配，结构如图3.1所示。

具体来说，为了保持和 ModelNet40 的物体的域尽可能一致，同时使得图片的质量更高，带上颜色、清晰的边缘等信息，我们从 ShapeNet 中挑选出类别与 ModelNet40 中一致的物体，将其按照不同视角渲染得到渲染模型图。

另外，我们还提出了另一种结构如3.1中的 c) 所示。该结构使用了一个预训练的点云的编码器（[21]），并利用点云的特征增强学习。在测试的阶段，将点云的特征与文本匹配的分数和点云投射图的图片特征与文本匹配的分数加权结合，得到最终的结果。

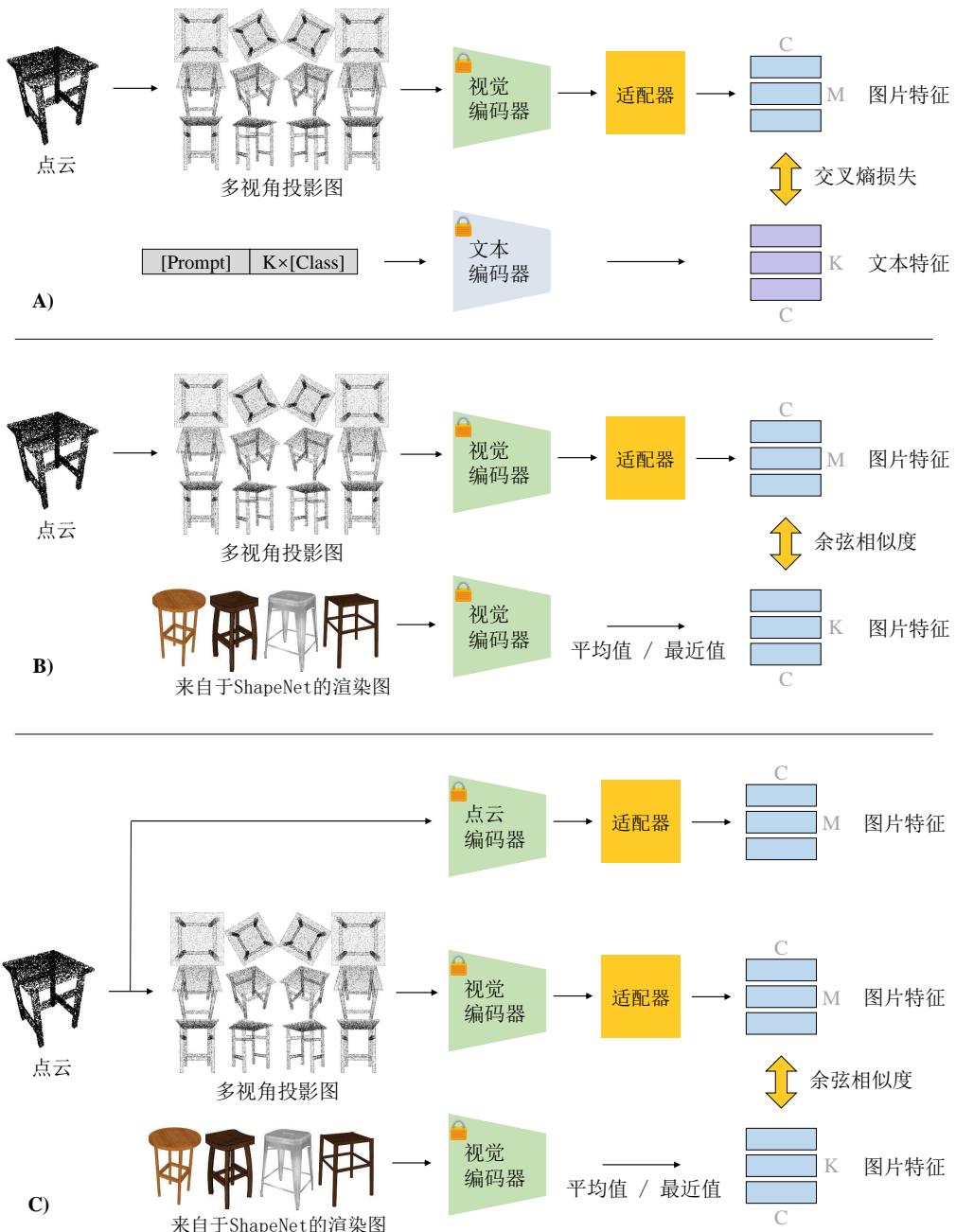


图 3.1 在语义层面的特征对齐的训练阶段示意图。A: PointCLIP 的少样本训练结构示意图, 将点云投影到 M 个投射图上, 使用在二维图片上预训练的 CLIP 进行分类, 并在最后加入交叉熵损失的监督; B: 收集来自于 ShapeNet 的渲染图, 并使得点云投射图的图像特征向量与渲染图的特征向量靠近; C: 加入一个预训练的点云编码器, 利用点云本身的几何特征增强学习。

3.2 在输入层面的特征对齐

我们认为，经过了 CLIP 预训练的视觉编码器之后的特征本身就是含有潜在语义信息的，在这样抽象的层面进行对齐是难以保持原有的泛化性的。在 CLIP 中，视觉编码器分为 ResNet^[29] 和 ViT^[4] 两种。给定图片 $I \in \mathbb{R}^{H \times W \times C}$ ，ResNet 中图片会先通过 stem 层，其中包含以下操作：

1. 一个卷积层：使用较大的卷积核（如 7x7），较大的步长（如 2）和较少的输出通道。这个卷积层可以捕获图像的一些基本特征，并降低输入图像的空间尺寸。
2. 批量归一化（Batch Normalization）：在卷积操作之后进行归一化，加速网络收敛并提高模型性能。
3. 激活函数（如 ReLU）：引入非线性激活函数，增强模型的表达能力。
4. 最大池化（Max Pooling）：进一步降低图像的空间尺寸，提取更抽象的特征。

此时我们会得到 $F \in \mathbb{R}^{F, H//4, W//4}$ 的特征图。由于前序操作的卷积层步长较大，因此我们可以认为这一步得到的特征依然保留了原来图片空间的形状特征。之后，ResNet 会将得到的特征进一步输入到后续的一系列带有残差连接的 BottleNeck 层。

另外一种视觉编码器 Vision Transformer ViT 先将 2D 图像分割成一系列扁平化的 2D 图像块 $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$ ，并使用一个卷积层将这些图像块转换成一维视觉嵌入的词例 $E_I(I) \in \mathbb{R}^{N \times D}$ ，其中 N 是词例的总数量， $P \times P$ 是图像块大小， D 是每个图像词例的维度。 H 和 W 是给定图像的高度和宽度，总图像块数量为 $N = HW // (P^2)$ 。之后，将位置编码加到图像块词例上，添加一个总的类别词例（<cls>），将这些一起送入 Transformer 以进行特征提取。从数学上讲，2D ViT 可以表示如下：

$$\begin{aligned}
 z_0 &= \left[x_{cls}, E_I(I_{1,1}), \dots, E_I(I_{\frac{H}{P}, \frac{W}{P}}) \right] + E_{pos} \\
 \tilde{z}_l &= MSA(LN(z_{l-1}) + z_{l-1}) \\
 z_l &= MLP(LN(\tilde{z}_l)) + \tilde{z}_l \\
 y &= H^{cls}(LN(z_L^0))
 \end{aligned} \tag{3.1}$$

其中 $E_I(\cdot)$ 是图像分词器模块，用于为每个图像块提取视觉嵌入的词例， x_{cls} 是类别词例。Transformer 由 L 层的层归一化 $LN(\cdot)$ ，多头自注意力 $MSA(\cdot)$ 和多层次感知器 $MLP(\cdot)$ 组成，每个块后都用残差连接。 H^{cls} 代表分类头，将最后一层

的 x_{cls} 特征作为输入。

我们希望将点云的浅层特征转换为可被 ResNet 的 Bottleneck 层编码的浅层特征向量，或是可被 ViT 中 Transformer 编码的图像块。因此，这一小节里我们将探讨如何在视觉编码器之前将输入层面的浅层特征对齐。

3.2.1 投射图像和渲染图像的对应问题

使用 PyTorch3D^[37] 的框架可以将原始 ModelNet 数据集中的物体渲染成为更接近真实图像的图片。为了给投影图引入一个引导，我们按照 PyTorch3D 中的代码实现重新计算了点云的投影方式。假设点云中某点的世界坐标为 $\mathbf{p}_w = (x, y, z)$ ，其齐次坐标为 $\mathbf{p}_w = (x, y, z, 1)$ ，我们希望计算该点在渲染图像上的坐标 $\mathbf{p}_c = (x, y)$ 。在 PyTorch3D 的渲染过程中，相机会首先将世界坐标通过变换矩阵 R 和 T 转换到相机坐标 \mathbf{p}_c 。我们可以通过以下式子来计算 \mathbf{p}_w 与 \mathbf{p}_c 之间的转换关系：

$$\mathbf{p}_c = P_{cam} \cdot \mathbf{p}_w = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \mathbf{p}_w \quad (3.2)$$

接下来，我们将相机坐标投影到图像平面上，可以定义如下的投影矩阵：

- fov_y : 垂直方向上的视场角 (field of view angle)
- r : 长宽比 (width / height)
- K :

$$K = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

其中， $a = \frac{1}{r \tan(\frac{fov_y}{2})}$, $b = \frac{1}{\tan(\frac{fov_y}{2})}$.

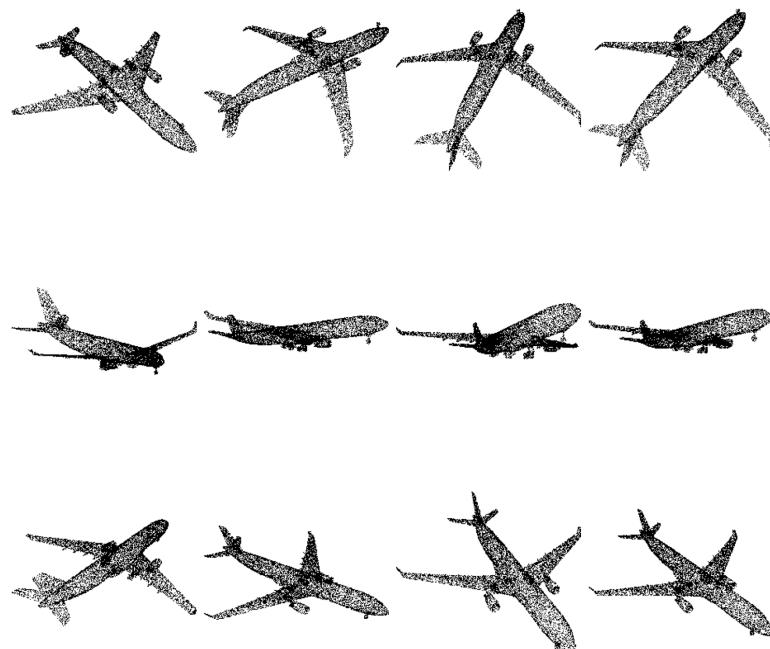
接下来，我们可以使用投影矩阵 K 将相机坐标点 P_{cam} 投影到图像平面上。首先，我们将其转换为归一化设备坐标 (NDC) P_{ndc} ：

$$P_{ndc} = K * P_{cam} / Z_{cam} \quad (3.4)$$

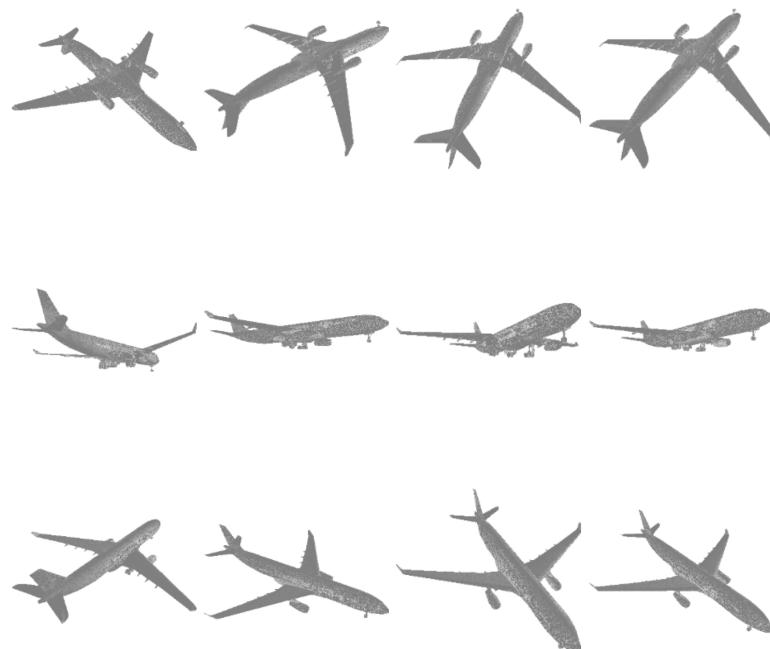
之后，即可以得到在图像上的坐标

$$\begin{aligned} x_{image} &= (P_{ndc}.x + 1) * w / 2 \\ y_{image} &= (1 - P_{ndc}.y) * h / 2 \end{aligned} \quad (3.5)$$

在图3.2中，我们展示了将物体随机旋转若干个角度后，分别得到其投影图和渲染图的结果。



(a) 将物体随机投影到若干个角度



(b) 将物体按同样角度渲染

图 3.2 投影-渲染示意图

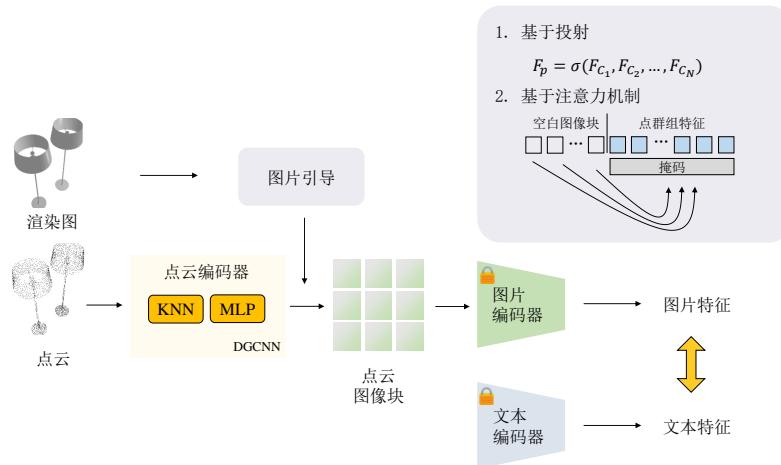


图 3.3 构造图像块的方法结构示意图

3.2.2 构造图像块

我们构造了如图3.3的结构。在图片编码器前端，我们引入了一个点云编码器，该点云编码器由 DGCNN^[13] 的结构改进而来。DGCNN 引入了一系列的边缘卷积 (EdgeConv) 层，该结构利用局部几何结构来动态地构建局部邻域图，并连接相邻点对的边上应用类卷积运算。给定输入的点云 $\mathcal{P} \in \mathbb{R}^{N \times F}$ 。在最初输入的时候， $F = 3$ ，即每个点仅由自己的坐标表示 $\mathbf{x}_i = (x_i, y_i, z_i)$ ，或也可以包括颜色、表面法线等其他的信息。而在 DGCNN 的网络架构中，每个后续的层都在前序的输出上进行操作，因此 F 也可以表示给定层输出的点特征的维度。在某一层中，DGCNN 首先利用 K 最邻近图 (k-NN) 来对每个点构建局部点云结构的有向图 $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ ，其中 $\mathcal{V} \in \{1, \dots, k\}$ 为最近的 k 个点， $\{E \subseteq \mathcal{V} \times \mathcal{V}\}$ 为边。之后，DGCNN 在每条边上计算边特征，聚合与点连接的所有边的特征来计算出某个点的特征向量。具体的数学表达如下：

$$\mathbf{e}_{ij} = h_\Theta(\mathbf{x}_i, \mathbf{x}_j), h_\Theta : \mathbb{R}^F \times \mathbb{R}^F \rightarrow \mathbb{R}^{F'} \quad (3.6)$$

$$\mathbf{x}'_i = \square_{j:(i,j) \in \mathcal{E}} \mathbf{e}_{ij} \quad (3.7)$$

其中 h_Θ 为参数可学习的非线性函数， \square 为聚合操作，可以是最大值或是平均值等。经过每一层边缘卷积后，语义上接近的点之间距离会更近，具体可视化的结果如图3.4。正是因为 DGCNN 可以为每个点提取特征，且经过 DGCNN 后每个点的特征包含一定的局部结构特征信息，因此我们选择它作为点云的编码器。

经过点云编码器之后，我们可以得到每个点的特征 $F \in \mathbb{R}^{N \times F}$ 。之后，我们分别提出了两种方式来获得点云的图像块。假设需要得到的图像块为 $P \in \mathbb{N} \times \mathbb{F}$ ，

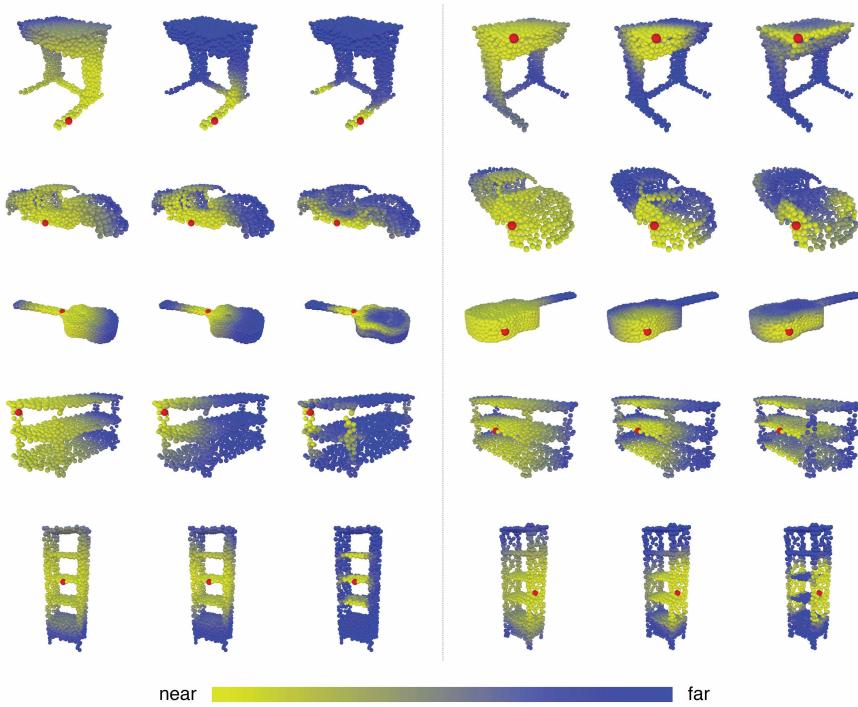


图 3.4 DGCNN 网络在不同阶段展现出的特征空间结构，可视化图片结果来自于 DGCNN 论文^[13]。可视化为红点与其他点之间的距离。左侧：在输入的 \mathbb{R}^3 坐标系中的距离；中间：在中间层的特征距离；右侧：在最后一层特征空间中的距离。可以看到，随着层数的加深，语义上相似的点在特征空间的距离会更近。

其中 N 为图像块的数量。我们可以用零初始化这些图像块。

首先我们考虑基于投影的方式。对于每个图像块的特征，可以写为表达式 $F_p = \theta(F_{C_1}, F_{C_2}, \dots, F_{C_N})$ 。其中 θ 可以为最大值操作、平均值操作或取视角最前的点等。当 $N = H \times W$ 时，相当于我们直接将每个点投射到图片上而非某个划分的图像区域上；得到的特征压缩到三通道，可以认为构造了一个类似于 p2p^[28] 的上色模块，它可以根据点云的结构信息为投射图赋予可被 CLIP 理解利用的颜色信息。

可学习性更强的方法则是构造一个基于注意力机制的方式。一般地，多头注意力机制可以写为如下式子：

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h) W^O$$

其中 $\text{head}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right) = \text{Attention}(Q_i, K_i, V_i)$, (3.8)

$$\text{Attention}(Q_i, K_i, V_i) = \text{softmax}\left(\frac{Q_i K_i^\top}{\sqrt{d_k}}\right) V_i$$

我们先在所有点的特征上加上位置编码，应用互相的自注意力机制，使得点之间可以建立联系；之后加入零初始化的图像块一起使用自注意力机制计算。另外，

由于图片块上的空间关系，我们在计算过程中引入了一个掩码。具体来说，每个零初始化的图像块只能接收到投射在该图像块上的点的信息，每个点只能接收到在同一图像块上的点的信息。

3.2.3 对比学习的策略

在 CLIP 模型的训练时，给定 N 对图像-文本，CLIP 需要在 $N \times N$ 种图像-文本对的组合中选择出真实配对的图像-文本对，即最大化 N 个匹配的图像和文本特征的余弦相似度，同时最小化 $N \times N - N$ 个不正确配对的特征的余弦相似度，使用这些相似度的分数来优化对称的交叉熵损失。在训练过程中，CLIP 的伪代码如图3.5所示。

```
# image_encoder - ResNet or Vision Transformer
# text_encoder - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]         - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t              - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T) # [n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss  = (loss_i + loss_t)/2
```

图 3.5 CLIP 训练过程中的伪代码，代码图来自于 CLIP 原论文^[5]。

为了保持 CLIP 本身的性质，我们认为，在训练中应该尽量保持 CLIP 的训练策略，一个简单的方法就是使用对比学习的损失，而不直接引入和标签的分类损失。

第四章 实验结果

4.1 在语义层面的特征对齐的实验结果

按照章节三中描述的方法，在 b) 中我们向 CLIP 的物体编码器后加入了一些多层感知器组成的适配器，从 ShapeNet 中挑选出一些 ModelNet10 分类中的样例，在训练时使得点云的特征与所选出图像的平均特征尽量靠近。我们发现使用平均特征、最近的图像特征在结果上没有很大的区别。测试时，我们直接使用上述 ShapeNet 上训练好的模型，测试 ModelNet10 上的分类性能。而在 c) 方法中，我们将点云和图像的模态一同引入，在点云模态上使用分类损失监督，同时增加点云和图像之间的 l_1 -loss 约束其靠近。通过这种方式，所获得的测试结果如表4.1所示。同时，我们在测试集上将经过适配器后的图像特征和标签文本特征进行匹配，可视化了二者之间的余弦相似度，结果如图4.1。训练均为从每个类中随机挑选 16 个样本的少样本学习。

准确率 (%)	b) 图片 特征对齐	c) 点云辅助 图片特征对齐	PointCLIP w/o searching	PointCLIP w searching
总计	71.16	91.52	88.66	89.45
浴缸 (bathtub)	75.00	98.00	98.00	98.00
床 (bed)	62.18	95.00	95.00	96.00
椅子 (chair)	85.61	99.00	96.00	96.00
办公桌 (desk)	61.00	84.88	87.21	86.00
衣柜 (dresser)	56.45	88.37	90.70	91.50
显示器 (monitor)	96.50	99.00	97.00	99.00
床头柜 (night stand)	35.37	66.28	41.86	53.37
沙发 (sofa)	62.50	98.00	99.00	99.00
桌子 (table)	93.75	87.00	90.00	90.00
厕所 (toilet)	83.33	98.00	100.00	100.00

表 4.1 语义层面对齐的实验结果

当只使用图像的时候，我们可以发现，由于没有在最终的分类任务上训练，模型的准确性有所降低；同时这种方法极受数据集偏差的影响。具体来说，ShapeNet 作为一个标注并不整齐的大数据集，其中类似于床头柜这样的分类包含的样本并

Cosine similarity between text and mean image features

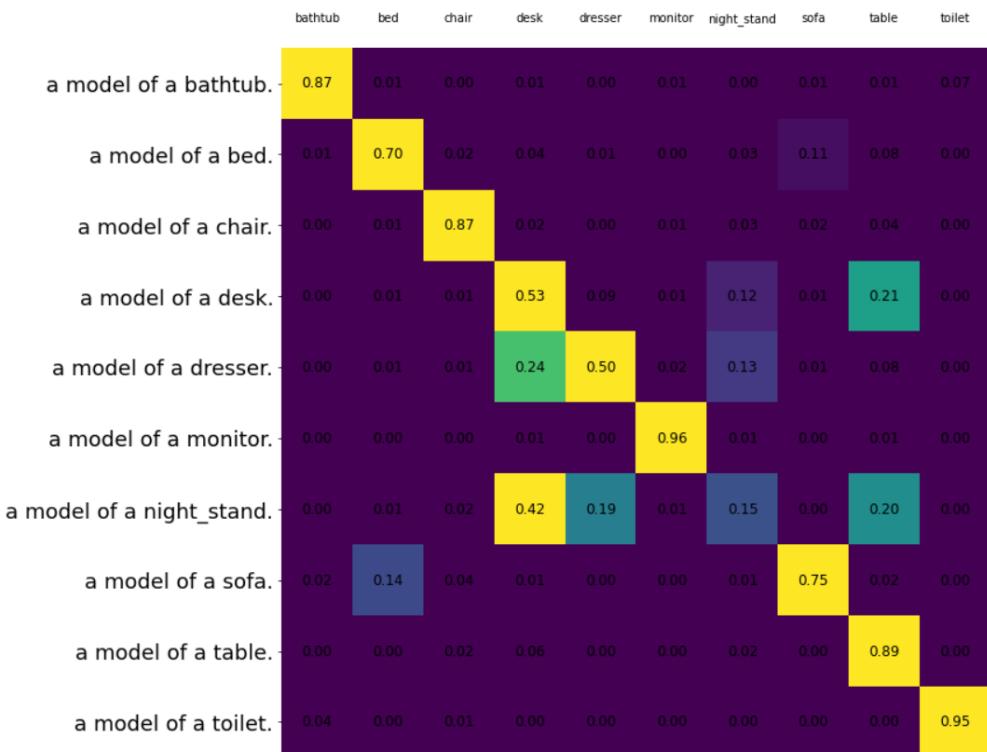


图 4.1 可视化语义层面对齐的图像-标签相似度

不多，且特征不明显，于是床头柜这一类上的性能显著地有所降低。例如桌子和办公桌、沙发和床这类相似的标签在 ShapeNet 的数据中有时候甚至会一起出现，即 ShapeNet 中对于这些概念的区分并不绝对，其语义上的概念也和 ModelNet10 上存在一定的差异，所以我们的方法在这些类上的表现也较低。

相对地，同时使用图像和点云的模块，使得两个模态的特征各自发挥作用，则确实会提高表现性能。我们的方法基本提升了每个类的准确率，在十分类的任务上取得了 SOTA 的性能。

然而，当划分基类和新类时，在语义层面对齐的做法在新类上的准确率都为零。这是由于从零初始化的适配器在训练过程中对分类的语义更加敏感，其学习的知识无法迁移到没有见过的例子上。

4.2 在输入层面的特征对齐的实验结果

按照章节三中描述的方法，我们先简单地基于投射的方法构造图像块，对模型进行 16 样本的训练。结果如图4.2所示。

	Base														Novel					
	airplane	bathtub	bed	bench	bookshelf	bottle	bowl	car	chair	cone	cup	curtain	desk	door	dresser	flower pot	glass box	guitar	keyboard	
Acc/1e	100	12	59	0	0	97	95	98	95	95	70	60	0	0	0	0	1	84	25	
Acc/100e	100	46	99	35	85	96	100	99	96	95	75	90	1.16	55	0	45	16	86	5	
Diff	0	34	40	35	85	-1	5	1	1	0	5	30	1.16	55	0	45	15	2	-20	

图 4.2 在输入层面基于投射的方法构造图像块的结果，表格第二行为训练一个 epoch 后的结果，第三行为训练 100 个 epoch 后的结果。

观察结果我们可以发现，经过训练之后，除了基类基本都有准确率的提升，部分新类上的分类准确率也有所上升。这表明模型学到了一部分结构层面的知识，并可以将其用于未见过的分类任务上。然而，对于一部分新类在训练后仍然无法分类正确，可能是因为在 CLIP 的图像空间中，这些类的域和 ModelNet 中的图像域差距较大，特征有较大差别，因此不具有很强的迁移性。

更进一步地，我们运用注意力的机制，并在训练过程中添加 CLIP 的对比损失。然而，简单地添加对比损失会导致模型完全学不到任何信息。如图4.3所示，在训练过程中，虽然损失函数的数值有所下降，但是模型的准确率一直十分糟糕。我们推测，对比损失需要大量的数据才有可能成功训练。为了解决这个问题，我们使用了在线的渲染器，实时地随机将点云旋转一个角度，理论上来说可以获得无限的投射图-渲染图-文本匹配对。然而，这种方法的显存占用量是巨大的。同时，我们查阅了相关对比训练的资料，发现除了使用大量级的数据量，在单次推理过程中，同一批次的数据量（batch size）也非常大。在目前的代码框架中，实现这样的训练是较为困难的。

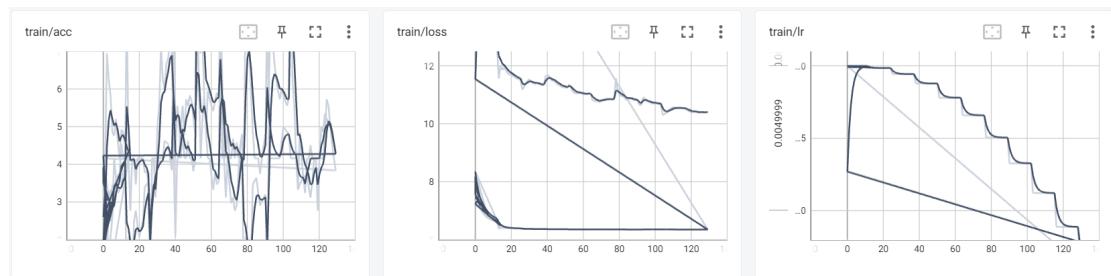


图 4.3 简单添加对比损失时的训练日志文件。左图为准确率的变化，中间图为损失函数值的变化，右侧为学习率调整的过程。

第五章 结论

在本文中，我们探讨了如何将 CLIP 模型应用于三维点云的识别。具体来说，我们分别采用了在特征维度和输入维度将不同模态对齐的方式，尝试将 CLIP 作为一个点云的编码器。在特征维度上进行对齐，我们取得了 SOTA 的性能，但仍然受到了无法泛化的制约。在输入层面对齐的泛化性有所提升，然而总体的准确率却并不好；而且引入对比学习的训练难度较大，导致没有得到一个泛化和准确率兼得的方法。

我们的实验表明将 CLIP 应用于三维领域是可行的。另外，近期一些有关于点云语义分割、场景中物体识别等其他的工作^[38–40] 也证明了将 CLIP 迁移到三维领域的可行性。对于未来的工作，我们认为可以分为如下几点：

- 运用多卡并行的技术，在渲染图-投射图-文本匹配对上进行大量的对比学习；或收集 ShapeNet 的图片和点云进行对比学习的预训练。通过这样的方式实现在输入层面将点云转为更可靠的分词。
- 融入大型语言模型，利用语言模型中丰富的知识引导学习，使得模型更能学习到物体具体细粒度的特征。
- 在其他的数据集或任务上继续测试，或运用训练好的预训练模型，实现一个通用的从二维图像迁移的三维基础模型。

参考文献

- [1] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
- [2] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners [J]. OpenAI blog, 2019, 1(8): 9.
- [3] RAFFEL C, SHAZEER N, ROBERTS A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. The Journal of Machine Learning Research, 2020, 21(1): 5485–5551.
- [4] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.
- [5] RADFORD A, KIM J W, HALLACY C, et al. Learning transferable visual models from natural language supervision[C]// International conference on machine learning: PMLR, 2021: 8748–8763.
- [6] XIE S, GU J, GUO D, et al. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding[C]// Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16: Springer, 2020: 574–591.
- [7] YU X, TANG L, RAO Y, et al. Point-bert: Pre-training 3d point cloud transformers with masked point modeling[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022: 19313–19322.
- [8] MA X, QIN C, YOU H, et al. Rethinking network design and local geometry in point cloud: A simple residual mlp framework[J]. arXiv preprint arXiv:2202.07123, 2022.
- [9] CHANG A X, FUNKHOUSER T, GUIBAS L, et al. ShapeNet: An Information-Rich 3D Model Repository: arxiv:1512.03012 [cs.gr] [R]: Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.
- [10] ZHANG R, GUO Z, ZHANG W, et al. Pointclip: Point cloud understanding by clip[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022: 8552–8562.
- [11] QI C R, SU H, MO K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation[C]// Proceedings of the IEEE conference on computer vision and pattern recognition, 2017: 652–660.
- [12] QI C R, YI L, SU H, et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space[J]. Advances in neural information processing systems, 2017, 30.
- [13] WANG Y, SUN Y, LIU Z, et al. Dynamic graph cnn for learning on point clouds[J]. Acm Transactions On Graphics (tog), 2019, 38(5): 1–12.

- [14] CHOY C, GWAK J, SAVARESE S. 4d spatio-temporal convnets: Minkowski convolutional neural networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019: 3075–3084.
- [15] PASZKE A, GROSS S, CHINTALA S, et al. Automatic differentiation in pytorch [Z], 2017.
- [16] WU W, QI Z, FUXIN L. Pointconv: Deep convolutional networks on 3d point clouds[C]// Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, 2019: 9621–9630.
- [17] SU H, MAJI S, KALOGERAKIS E, et al. Multi-view convolutional neural networks for 3d shape recognition[C]// Proceedings of the IEEE international conference on computer vision, 2015: 945–953.
- [18] FENG Y, ZHANG Z, ZHAO X, et al. Gvcnn: Group-view convolutional neural networks for 3d shape recognition[C]// Proceedings of the IEEE conference on computer vision and pattern recognition, 2018: 264–272.
- [19] GOYAL A, LAW H, LIU B, et al. Revisiting point cloud shape classification with a simple and effective baseline[C]// International Conference on Machine Learning: PMLR, 2021: 3809–3820.
- [20] HAMDI A, GIANCOLA S, GHANEM B. Mvtn: Multi-view transformation network for 3d shape recognition[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021: 1–11.
- [21] WANG H, LIU Q, YUE X, et al. Unsupervised point cloud pre-training via occlusion completion [C]// Proceedings of the IEEE/CVF international conference on computer vision, 2021: 9782–9792.
- [22] PANG Y, WANG W, TAY F E, et al. Masked autoencoders for point cloud self-supervised learning[C]// Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II: Springer, 2022: 604–621.
- [23] ZHANG R, GUO Z, GAO P, et al. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training[J]. arXiv preprint arXiv:2205.14401, 2022.
- [24] AFHAM M, DISSANAYAKE I, DISSANAYAKE D, et al. Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022: 9902–9912.
- [25] LI Z, CHEN Z, LI A, et al. Simipu: Simple 2d image and 3d point cloud unsupervised pre-training for spatial-aware visual representations[C]// Proceedings of the AAAI Conference on Artificial Intelligence: volume 36, 2022: 1500–1508.
- [26] JING L, CHEN Y, ZHANG L, et al. Self-supervised modal and view invariant feature learning [J]. arXiv preprint arXiv:2005.14169, 2020.
- [27] WANG Y, FAN Z, CHEN T, et al. Can we solve 3d vision tasks starting from a 2d vision transformer?[J]. arXiv preprint arXiv:2209.07026, 2022.

- [28] WANG Z, YU X, RAO Y, et al. P2p: Tuning pre-trained image models for point cloud analysis with point-to-pixel prompting[J]. arXiv preprint arXiv:2208.02812, 2022.
- [29] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 770–778.
- [30] VASWANI A, SHAZEE N, PARMAR N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
- [31] RAMESH A, DHARIWAL P, NICHOL A, et al. Hierarchical text-conditional image generation with clip latents[J]. arXiv preprint arXiv:2204.06125, 2022.
- [32] ROMBACH R, BLATTMANN A, LORENZ D, et al. High-resolution image synthesis with latent diffusion models [Z], 2021.
- [33] XU H, GHOSH G, HUANG P Y, et al. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding[C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, 2021.
- [34] WANG M, XING J, LIU Y. Actionclip: A new paradigm for video action recognition[J]. arXiv preprint arXiv:2109.08472, 2021.
- [35] GUZHOV A, RAUE F, HEES J, et al. Audioclip: Extending clip to image, text and audio[C]// ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): IEEE, 2022: 976–980.
- [36] WU H H, SEETHARAMAN P, KUMAR K, et al. Wav2clip: Learning robust audio representations from clip[C]// ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): IEEE, 2022: 4563–4567.
- [37] RAVI N, REIZENSTEIN J, NOVOTNY D, et al. Accelerating 3d deep learning with pytorch3d [J]. arXiv:2007.08501, 2020.
- [38] PENG S, GENOVA K, JIANG C, et al. Openscene: 3d scene understanding with open vocabularies[J]. arXiv preprint arXiv:2211.15654, 2022.
- [39] DING R, YANG J, XUE C, et al. Language-driven open-vocabulary 3d scene understanding[J]. arXiv preprint arXiv:2211.16312, 2022.
- [40] YANG J, DING R, WANG Z, et al. Regionplc: Regional point-language contrastive learning for open-world 3d scene understanding[J]. arXiv preprint arXiv:2304.00962, 2023.

致 谢

本文大部分工作完成于大四上学期 CVPR2023 的前夕。彼时在三维领域磕磕绊绊从视觉定位到迁移学习一路走来，同时在为升学和语言考试焦头烂额，在过程中遇到了无数困难和挫折。这是我第一次有幸参与一段扎实的科研研究，虽然最后结果并不令人讨喜，但仍然在过程中收获了很多很多。对我来说，科研是一个持续的过程，在过程中永远没有一个确定的答案和一个正确的方法。然而这个过程本身就是迷人的：大家在各种实验的蛛丝马迹中探寻出一条清晰的逻辑链，并一点一点地启发着别人一起探寻科技发展的前沿。这本身就是一件令人激动的神奇的事情，同时在这个过程中，也少不了无数贵人相助，无法一一表达感激之情。

感谢我的导师杨思蓓教授在过程中一直的支持与帮助。在刚刚进入课题组之时，虽然确实处于“什么都不会”的阶段，但老师十分认真地带领我进入了课题组的工作。她从最开始引导我们如何读论文，到如何形成自己的想法，再到做课题过程中的思路方向，其中老师付出的时间和心血不计其数。能在科研的最开始得到如此悉心的指导和大量的交流，是极为宝贵的机会。她会关注组里每一个学生的特点，给予不同的指导和帮助，真正认真地教导年轻学生，其中令人感动的瞬间不计其数。不仅如此，老师在人生上也是我的导师和榜样。在申请学校和未来选择的问题上，老师几次和我长谈，给出了十分中肯且真诚的意见；而在科研的宏观视野上，老师对于研究的理想和热情时时刻刻在感染着我，她对于领域前沿发展的见解也非常具有启发性。正是因为老师的影响，我越来越热爱目前的研究方向，充满了对这个领域的憧憬和信心。

感谢和合作者杨斌同学的缘分，从大二开始，我们在各种专业课上的组队合作，讨论学业和作业问题，而之后又一同进入课题组分工做项目。同时在生活上我们也是非常知心的朋友，我们一起学习和摸鱼、一起去看展和吃饭，这构成了我一大部分大学学业生活的美好回忆。感谢组里石骋学长、唐嘉晋学长和黄涵卓学长的帮助，他们在项目的各种问题上和我一起讨论，提供了不少指导与照顾。感谢实验室同组的朋友们，大家一直对我的升学申请和项目十分关心，即便在我不出现在实验室的一段时间里也经常带我一起吃饭聊天，鼓励我度过最忙、压力最大的时候。感谢贺治鹏同学、张心怡同学和我的室友们，他们在大学四年里的很多个日子里和我一起互相分享生活，我从未想过自己也可以在上科大建立

起这样的朋友圈，在每个节点总可以得到很多人的支持。同时感谢我的高中同学叶芝衡同学，即便在大学见面的次数屈指可数，我却在线上可以一直与她洽谈一如当初，她给了我很多宝贵的建议。

感谢我的父母给我提供各种方面的支持，他们让我可以放心地去做自己喜欢的事情，是永远的最坚实的后盾。

大学的四年在我人生的二十几年来说或许算不上是一个高光时刻，和贺治鹏回想大一大二的时候似乎都过得很痛苦，那时候觉得自己什么也学不会，什么也搞不懂，而到申请季也总觉得自己的大学什么都没有做。总觉得高中的环境下成绩一直还不错，可高中的素质教育反而让我在大学“躺”了很多，封闭的社交圈、缺少的冲劲和不够扎实的基础。同时，似乎也总觉得在上科大的精神生活十分贫瘠，觉得自己好像活的越来越普通。然而，为了分析 *thanatopsis* 诗歌去翻阅哲学书籍的是那个小姑娘，一直在坚持摄影和观察的是那个诗意的小姑娘，开始钻研艺术史开始艺术创作的也还是那个会对哲学文学感兴趣的小姑娘。是在大学四年这里更进一步地认识了自己，思考了自己和社会、和不同人的关系；更重要的是，现在的我正在做自己热爱的事情，在慢慢寻找自己想要的轨道。所以，存下与大家的回忆和大家地鼓励，继续勇敢地面对生活的一切吧！