电 子 科 技 大 学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

专业学位硕士学位论文

MASTER THESIS FOR PROFESSIONAL DEGREE

论文题目 **基于深度学习的点云分割技术研究**

专业学位类别	工程硕士
学　　号	**201922090628**
作者姓名	杨泽宇
指导教师	饶云波　　副教授
学　　院	信息与软件工程学院

分类号 TP311.5 密级

UDC 注1 004.41

学 位 论 文

基于深度学习的点云分割技术研究

（题名和副题名）

杨泽宇

（作者姓名）

指导教师 饶云波 副教授

电子科技大学 成 都

（姓名、职称、单位名称）

申请学位级别 硕士 专业学位类别 工程硕士

专业学位领域 软件工程

提交论文日期 2022 年 3 月 10 日 论文答辩日期 2022 年 5 月 17 日

学位授予单位和日期 电子科技大学 2022 年 6 月

答辩委员会主席

评阅人

注 1：注明《国际十进分类法 UDC》的类号。

Research on point cloud segmentation technology based on deep learning

A Master Thesis Submitted to

University of Electronic Science and Technology of China

Discipline **Master of Engineering**

Student ID **201922090628**

Author **Zeyu Yang**

Supervisor **Prof. Yunbo Rao**

School **School of Information and Software Engineering**

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

作者签名：杨泽宇　　日期：2022 年 5 月 30 日

论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文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、汇编学位论文。

（保密的学位论文在解密后应遵守此规定）

作者签名：杨泽宇　　导师签名：倪云波

日期：2022 年 5 月 30 日

摘 要

随着人工智能的发展，计算机视觉得到了前所未有的研究和应用。三维点云分割作为计算机三维场景理解的基础，已经成为近年来的研究热点。精准的点云语义预测是自动驾驶、医疗检测、智慧城市等领域的关键技术，有着重要的应用意义。随着点云数据的丰富和点云算法的广泛研究，目前大多数点云网络已解决无序性、非结构化等挑战，但仍然存在语义预测不够精确，场景数据利用不完全等问题。本文围绕上述问题，开展基于融合特征和基于注意力机制的点云语义分割算法研究。研究内容包括融合 RGB 特征的点云分割算法，基于 Transformer 的点云学习网络和基于通道自注意力的点云分割网络，主要内容和贡献如下：

（1）针对场景数据利用不充分，场景分割的准确率低等问题，提出了融合 RGB 特征的点云语义分割网络。该算法将图像 RGB 特征根据空间投影融合到点云数据中。通过点云相对位置信息提取潜在的语义信息，设计新颖的交叉空间注意力模块，搭建分割网络整体架构。算法在 ScanNet 数据集上达到了 68.2%的分割效果，相比于 PointNet++提高了 13.7%，验证了该算法对场景数据的精准分割。

（2）针对点云学习泛化能力差，特征语义无法精准提取的问题，提出了基于 Transformer 的点云学习网络。该算法利用 Transformer 中的自注意力机制，设计了适应点云的自注意力结构，并引入特征位置编码，分别对点云的特征内关联性和特征之间关联性进行学习。网络在 ShapeNet 数据集上 mIoU 值为 84.2%，相比于 PointNet 提高了 0.5%。在 ModelNet40 数据集上分类精度达到 93.3%，S3DIS 数据集上 mIoU 取得 60.6%的成绩，相比于 PointNet++分别提高了 2.6%和 6.1%，证明了本章网络具有较强的点云分类能力和分割能力。

（3）针对自注意力机制在点云学习中计算效率低的问题，设计了基于通道自注意力的点云分割网络，优化注意力结构使得网络计算效率得到提高。此外，为进一步有效学习局部区域的点云特征，采用基于余弦距离的 K 邻近算法对点云特征进行分组，设计了点云局部特征抽象模块，使网络可以掌握更加丰富的特征语义信息。网络在 ShapeNet 数据集上取得了 85.9%的成绩，较基于 Transformer 的点云学习网络和 PointNet++高出 1.7%和 0.8%，证明了本网络在分割任务上具有较强的预测能力和泛化能力。

关键词：点云分割，点云分类，注意力机制，深度学习

ABSTRACT

With the development of artificial intelligence, computer vision has received unprecedented research and applications. 3D point cloud segmentation, as the basis for computer 3D scene understanding, has become a research hotspot in recent years. Accurate point cloud semantic prediction is a key technology in the fields of autonomous driving, medical testing, and smart cities, and has important application significance. With the enrichment of point cloud data and the extensive research of point cloud algorithms, most point cloud networks have solved the challenges of unorder and unregular, but there are still problems such as inaccurate semantic prediction and incomplete utilization of scene data. Focusing on the above problems, this thesis conducts research on point cloud semantic segmentation algorithm based on fusion features and attention mechanism. The research contents include point cloud segmentation algorithm fused with RGB features, Transformer-based point cloud learning network and point cloud segmentation network based on channel self-attention. The main contents and contributions are as follows:

(1) Aiming at the problems of insufficient utilization of scene data and low accuracy of scene segmentation, a point cloud semantic segmentation network fused with RGB features is proposed. The algorithm fuses image RGB features into point cloud data according to spatial projection. The potential semantic information is extracted from the relative position information of the point cloud, and a novel cross-spatial attention module is designed to build the overall architecture of the segmentation network. The algorithm achieves a segmentation effect of 68.2% on the ScanNet data set, and the test set effect is improved by 13.7% compared with PointNet++, which verifies the accurate segmentation of the scene data by the algorithm.

(2) In view of the poor generalization ability of point cloud learning and the inability to accurately extract feature semantics, a Transformer-based point cloud learning network is proposed. The algorithm uses the self-attention mechanism in Transformer to design a self-attention structure that adapts to the point cloud, and introduces the feature position encoding to learn the intra-feature correlation and the inter-feature correlation of the point cloud respectively. The network has an mIoU value of 84.2% on the ShapeNet dataset, which is 0.5% higher than that of PointNet. The classification accuracy on the

ModelNet40 dataset reaches 93.3%, and the mIoU on the S3DIS dataset achieves a score of 60.6%, which is 2.6% and 6.1% higher than that of PointNet++, which proves that the network in this chapter has strong point cloud classification and segmentation capabilities. .

(3) Aiming at the low computational efficiency of the self-attention mechanism in point cloud learning, a point cloud segmentation network based on channel self-attention is designed, and the attention structure is optimized to improve the computational efficiency of the network. In addition, in order to further effectively learn the point cloud features of the local area, the K-proximity algorithm based on cosine distance is used to group the point cloud features, and the point cloud local feature abstraction module is designed, so that the network can grasp more abundant feature semantic information. The network achieved a score of 85.9% on the ShapeNet dataset, which is 1.7% and 0.8% higher than the Transformer-based point cloud learning network and PointNet++, which proves that the network has strong prediction ability and generalization ability in segmentation tasks.

Keywords: Point cloud segmentation, Point cloud classification, Attention Mechanism, Deep Learning

目 录

图目录

表目录

第一章 绪论

1.1 引言

随着对深度学习技术的快速发展与广泛研究，人工智能也成了高新技术行业的重点发展方向。计算机视觉任务作为人工智能领域的重点研究方向，已涌现出不少精彩的成果。计算机二维视觉任务在以卷积式神经网络为代表的深度学习方法的迅速发展下取得了不菲的成绩。而对计算机三维视觉任务来说，特别是基于点云数据的深度学习方法的研究，却才刚刚步入高速发展的阶段。

在生产生活中，人们渴望机器可以像人一样收集现实世界的信息，客观处理和分析来源于现实世界的数据，从而给出相应的判断，代替人类或帮助人类完成某些活动。语义分割便是实现上述任务的重要技术，其所要求的精度也日益提升。点云语义分割是三维场景理解的基础和前提，点云分割是通过一定规则将点集中的点划分为不同的类别，预测为不同的语义标签。点云分割经过长时间的发展，研究者已经提出了大量的传统分割算法，然而传统分割算法受限于需要手工设置特征描述符，泛化能力较差。近年来，深度学习由于本身较强的函数拟合能力，以及语义信息学习能力，在各类视觉任务上得到了快速的发展。由于点云的无序性和稀疏性等特点，基于三维点云的场景分割一直是研究的难点。

根据点云语义分割的任务，可以将点云语义分割划分为场景语义分割、部分分割和实例分割[1]。场景中分类不同类别物体的像素即是场景语义分割，例如房间中的桌子和椅子，窗户和墙面；部分分割是语义分割的延伸，分割某个物体中的各个部分，因此也可称作零件分割；实例分割需要对场景中每一个物体预测出其类别，并区分出同一类别中的不同实例对象。

面对三维数据的日益庞大，三维点云逐渐受到了广泛的研究，涌现出越来越多的点云分割的方法。传统的分割算法泛化性差，计算代价高，点云信息无法充分提取，而基于深度学习的分割算法没有充分提取点云中点与点的关联关系。因此，如何从海量的点云信息中去除无用的信息，获得更多有价值的特征或关联信息仍然是三维点云分割研究的重点。

1.2 研究背景与意义

得益于三维建模技术以及 3D 传感器的飞速革新和广泛应用，三维点云数据精度不断提高[2]，三维点云数据开始了爆发式增长。基于点云的计算机视觉研究也取

得了较大的突破，进一步开启了许多基于三维点云的应用，如三维重建[3]、无人驾驶[4]、智能机器人[5]、文化遗产保护[6]、虚拟现实[7]和增强现实[8]等。在地理测量和建模中，研究人员使用三维点云数据来实现场景的三维重建[9]。在智能机器人领域中，三维点云数据是机器人感知外界场景[10]的基础空间信息。在无人驾驶领域中，点云数据被用来进行识别检测和分割道路目标[11]等任务。在智能城市领域中，城市建模、地形构建、建筑识别等[12]任务也常常用到三维点云数据。在文化遗产保护领域中，点云数据是文物重建和修复的数字化载体[13]。增强现实技术通过感知和理解 3D 几何形状，在正确的位置上显示虚拟影像。

三维点云数据拥有更丰富的信息。真实世界中物体的几何结构是三维的，三维点云恰好在几何结构上和真实物体一致，可以更好地刻画现实物体。并且同二维图像相比，点云包含更多的特征信息[14]，包括三维空间的隐形关联性和本身具有的几何颜色信息，比二维图像的信息量更大。三维点云数据由于本身的无序性和转换不变性，整体数据的语义不会受到空间刚性变换的影响。

1.3 国内外研究现状

随着深度学习的出现，人工智能领域中的许多任务都有了快速的发展，二维图像的语义分割技术在卷积神经网络的发展下逐渐成熟[15]。卷积神经网络在二维图像的视觉研究，如分类、分割和识别等领域取得了重大的突破，卷积神经网络也成为深度学习领域一项基础技术。与此同时，三维点云的识别和分割也开始逐渐从传统方法向深度学习方法发展。虽然卷积神经网络在计算机二维视觉任务上取得了优异的成绩，但是它不能直接应用于点云这种非结构化数据，这是由于点云是空间点的无序集合，集合中的空间点不具有二维像素的有序性和稠密性。近年来，研究者们为了克服这个问题，提出了许多方法，这些方法总体上可以分为基于规则化的方法和基于原始点云的方法。此外，在基于原始点云的方法中，包含基于多特征的方法和基于注意力机制的方法。

1.3.1 基于规则化的方法

随着深度学习的发展，三维点云数据已经逐渐被深度学习方法所青睐[16-18]。近些年来，深度学习技术在三维数据上开展了大量的研究和分析工作[19]。点云由于其无序性和离散型，无法直接作为规则化数据进行卷积操作，最直接的方法就是将点云转化为规则的结构[17]，然后输入卷积神经网络进行处理。点云是一种离散的三维数据，转换为规则化数据最直接的思路就是将其投影为图像，此类方法为基于

多视图的方法。同时，有些研究人员将点云转换为规则的体素数据[20]，体素数据具有类似图像像素的稠密性，这类方法称为基于体素的方法。

VoxNet[21]首先将原始的三维点云转换为规则排列的三维体素数据，并采用三维卷积神经网络对体素数据进行特征提取。文献[22]将深度图转换为体素表示，在体素数据上学习层次特征表达，完成形状识别的任务。由于体素的卷积形式为三维，计算量较大，为减小体素化方法的计算量，Li 等人[23]尝试将三维点云表示为体积场，并对场中数据进行自适应感知，减少了网络处理的数据量，最后将优化后的体积场输入到网络中进行处理。为了解决体素化方法受限于网格空间分辨率的问题，研究人员提出了八叉树[24]和 KD-Tree[25]结构，对点云三维空间进行划分，通过数据结构的方法对网络效率进行优化。文献[26]在八叉树结构的基础上，提出了 OGN-Conv 结构，降低了在高分辨率输出时体素的计算复杂度。国内研究人员对在八叉树上的三维卷积进一步优化，设计了一个可插拔的八叉树卷积模块[27]。PointGrid[28]方法将离散的点与网格结合起来，精简了网格大小，在空间和时间上的优化均有一定的进步。

MVCNN[29]是将多视图方法应用于点云识别的先驱，对多个视图进行特征提取，并采用最大池化得到整体的特征。Qi 等人[30]对多视图和体素的方法进行分析，对上述两种方法分别进行改进，提出了多向异性核和多分辨率方法。由于多视角方法会受到视角和遮挡的影响，Lawin 等人[31]采用投影的方法，将点云投影为二维图像，对二维图像进行语义分割，将图像中像素的分割结果聚合为点云中每个点的语义预测。为进一步提高分割表现，Kundu 等人[32]提出了虚拟视图的概念，将二维场景流扩充为虚拟多视图数据，在训练前生成虚拟视角的图片，并在虚拟视图数据上训练网络以进行分割。

1.3.2 基于原始点云的方法

3D 体素化和多视角方法均有一定的局限性[16]，体素化方法受限于体素的分辨率，过高的分辨率要求巨大的内存和计算资源，分辨率过低会导致体素无法包含足够的语义信息；多视角分割受限于视角和遮挡等问题，在投影过程中也存在信息损失的情况。为了充分利用三维点云的空间信息，研究人员尝试抛弃对点云的规则化手段，直接对点云进行端到端的学习。

PointNet[33]是首个可以直接处理离散无序点云的网络，该网络通过多层感知机提取点云的内在语义特征，并使用最大池化层得到点云的全局特征，采用对称函数解决点云的无序性问题。然而 PointNet 只关注各个点以及整体的特征，缺乏不同

层次下的局部特征。随后原作者引入层次化思想，通过对点云进行局部采样和分组，将 PointNet 作为不同层次的特征提取模块，设计了更加出色的 PointNet++[34]模型。

自 PointNet 和 PointNet++之后，越来越多的点云学习网络涌现出来，并且更加重视局部特征的提取。ShellNet[35]通过将点云分为多个尺度的球壳并逐壳卷积来提取点云特征。为了提高点云网络的效率和实时性能，Hu 等人[36]在大规模点云上探索出一种轻量的点云分割方案，在点云关键点选取中使用随机采样，并且通过注意力池化对特征进行更有效的提取。此外该网络无需对点云场景进行分块处理，大大增强了整个场景的语义信息。SPG[37]将点云组织为超点，超点带有丰富的边特征，通过 PointNet 对超点进行嵌入，最后将图卷积应用在边特征提取上，得到丰富的语义信息。DGCNN[38]提出了动态边卷积，将点云的关联性视为图的边，对边进行层次化动态卷积，逐步构建点云的关联性特征。Thomas 等人[39]提出了一种基于核点卷积（KPConv）的点云网络，点云中每个点通过核函数生成权重矩阵，球体内点云的特征由整个球体内点特征累加而来。Hua 等人[40]设计了一种可应用于每个点的卷积算子，并且中心点的特征由卷积算子与邻域点特征加权得到。在上述提取无序点云特征的网络之外，RSNet[41]将点云映射到带有序列信息的特征向量上，通过双向 RNN 提取序列特征。

在国内，上海交通大学的团队提出了 PointSIFT[42]模块，该模块对球查询和 KNN 算法进行了分析，对空间进行八个卦限的划分来分组点云，并依次对三个空间方向的点云进行卷积。山东大学团队提出了以 X-变换为核心的 PointCNN 网络模型[43]，通过 X-变换使得卷积可以直接应用在点云数据上。PointWeb[44]由香港中文大学团队和腾讯实验室共同提出，该算法提出了一个特征自适应模块，可以更好地挖掘局部区域的上下文信息，学习到点与点的相互作用信息。

1.3.3 基于融合图像的方法

面对三维网络的问题和挑战，研究人员期望在二维视觉上找到可适用三维数据的方法。得益于卷积神经网络的研究，在二维图像上的分割方法已经被广泛地发掘，越来越多优秀的方法被研究人员提出。自 2012 年 AlexNet[45]网络在 ImageNet ILSVRC 比赛惊艳众人后，卷积神经网络便成为了研究的热门，逐渐成为图像分割方法的主力军，也带领深度学习在计算机视觉领域中开始起步，得到了长远的发展。卷积层、池化层、全连接层和输出层组成了完整的卷积神经网络[46]，其中卷积层和池化层是提取特征的关键，将两者重复堆叠可以增强网络提取特征的能力。2014 年，VGG[47]对卷积核大小进行了详细的研究，采用卷积核较小的卷积层来增加网络的深度，并且逐渐成为图像处理的主干网络，被广泛用于图像分类、图像分割、

目标识别和检测等图像任务中。随着网络深度的增加，网络的特征学习能力不断增强，但同时网络的参数量也急剧增加，网络权重更新变慢，并且网络梯度会出现消失或爆炸的情况，导致网络越来越难以训练。为解决该问题，何凯明提出了ResNet[48]网络，该网络通过残差连接补充了原有特征的信息，避免了多层网络后梯度的消失或爆炸，可以成功训练高达 152 层的网络结构。

在点云学习网络飞快发展的同时，为进一步利用所有可用信息，研究人员着眼于二维数据和三维数据的语义结合，充分利用两种数据是改善当前视觉任务的一种途径。许多研究者通过将两种信息进行空间映射来解决视觉问题，3DMV[49]将 2D 图像特征映射到 3D 空间体素网格中，每个体素网格对应多个视图，每个体素特征通过池化对应视图的特征得到。Liang 等人[50]提出了一种将图像特征投影并融合到 BEV 特征中的方法，网络需要对两种数据进行特征提取。MVPNet[51]借鉴了前者的做法，将场景图像特征融合到点云数据中，并对融合后的点云数据进行语义分割。FuseSeg[52]在深度图分割过程中，将对应的 RGB 图像特征融合到深度图特征上完成点云的分割，充分利用了场景的二维数据和深度数据。同样的，在多视图方法中，有研究者将三维点云投影到二维平面上，对不同二维平面上的视图进行特征提取和融合，进而预测每个点的语义标签。如何将两种数据有效的结合，并且提高点云分割网络的精度是非常有挑战性的。

1.3.4 基于注意力机制的方法

近年注意力机制在点云学习中逐渐成为研究热点，点云网络逐渐开始采用注意力机制对特征进行提取。为了更好地捕捉点云的空间分布，LSA[53]采用多层感知机计算特征的相似度，设计了 SDW 模块通过特征相似度对点云特征进行更高级别的学习和语义修正。RandLA-Net 在池化部分同样采用多层感知机训练注意力分数矩阵，并与池化层组成了注意力池化模块。Zhao 等人[54]提出了一种基于注意力的得分细化模块，通过卷积学习到每个点的语义分数，采用最大池化对邻近点的特征进行聚合，将语义分数应用在聚合特征上得到最终分割结果。

Transformer[55]网络是自注意力模块的开创者，利用自注意力机制提取语言序列中词与词之间的关联性，在语义识别[56]、机器翻译[57]等自然语言学习任务[58]上取得了非常优异的成绩。近年来研究人员对 Transformer 在计算机视觉领域中进行了推广，涌现出许多成果，例如 ViT[59]实现了在二维图像上的分类，VisTR[60]在视频分割上取得了良好的成绩。

借助自注意力机制对关联性的提取，Transformer 在点云分类分割上被证明具有较强的学习能力。Point Transformer[61]通过 Transformer 机制设计了 SortNet 提取

点云特征，并聚合局部和全局特征来对点云进行预测。Point Cloud Transformer[62]（PCT）在自注意力的基础上设计了偏移注意力模块以优化网络权重，并通过邻域嵌入扩大网络处理点云的感受野。3DMedPT[63]采用 Lambda Attention 替换基本的自注意力结构，减小了模型计算的复杂度，并设计了局部上下文聚合模块来将点的局部和邻域信息充分利用。文献[64]提出一种 transformer-conv，区别于传统自注意力的结构，该结构利用点云的坐标和特征之间的关系进行计算，得到最终的特征。Pointformer[65]设计了以 U-Net 为整体结构的网络，采用自注意力机制学习并聚合局部和全局上下文特征，并且提出了一种通过自注意力机制计算关键点的算法。

1.4 主要研究内容及本文结构安排

本文由六个章节组成，研究内容及组织结构如下：

第一章：该部分介绍了本文的研究背景和意义，从点云分割方法分类上对国内外研究现状进行了详细阐述，并总结本文主要研究内容，概括本文后续章节安排。

第二章：本章总结了与本研究相关的理论知识和相关技术。首先对点云与图像的坐标系转换进行了讲解。其次分析了图像和点云数据的特点，分别介绍了近几年在点云语义分割领域中优秀的网络算法。最后阐述了注意力机制的思想，介绍了 Transformer 的简要结构。

第三章：本章对场景中的数据进行了分析，为了将场景数据中图像数据和点云数据充分利用并进行分割，提出了一种融合 RGB 特征的点云分割网络。首先提出了融合 RGB 特征的网络整体架构，将网络分为图像特征提取和点云语义分割两个分支，在点云空间坐标与像素平面坐标转换的基础上，提出了将像素特征学习到点云中的融合算法。其次，在 PointNet++网络的基础上，经过不断重构和优化，设计了相对特征提取模块和点云交叉空间注意力模块，并重新搭建一个点云场景语义分割网络。最终在场景分割任务进行实验评估，证明本章算法具有较强的分割能力。

第四章：本章提出了基于 Transformer 的点云学习网络，常见的点云学习网络无法较好地学习点云中点与点的关联关系，本章算法在点与点关联性提取的问题上进行了尝试，主要利用 Transformer 的自注意力机制，将点云的特征内关联性和特征之间关联性相结合，使网络既能对点云关联特征进行学习，也可以对局部特征进行语义修正。首先提出了一种利用自注意力机制的点云分类网络，并在此基础上进行优化，引入了特征位置编码，通过处理计算特征空间的特征距离来将相似特征进行语义学习，同时对自注意力机制的结构和正则化方式进行讨论分析以优化网络。最后将优化后的网络在点云分类、部分分割、场景语义分割进行了综合实验，

并通过对比其他优秀算法和消融实验，证明本章算法在点云学习领域中的优越性和可靠性。

第五章：为改进注意力机制在点云处理中的性能和效率，本章提出了基于通道自注意力的点云分割网络，在解决点与点关联性提取的问题上，进一步探索注意力机制的方法，优化了自注意力的性能和效率。通过改进注意力结构使得计算效率得到提高，并采用基于余弦距离的 K 邻近算法，结合点云的层次化思想，设计了点云局部特征抽象模块，使得网络可以掌握更加丰富的特征语义信息。最后构建整体网络模型，在点云部分分割数据集上进行实验，在精度和性能上进行对比分析。本章同时对比了 K 邻近算法参考不同距离的影响，检验并证明了本章算法的正确性和精确性。

第六章：对本文进行概括总结，并对后续研究工作提出展望。

第二章 相关理论与技术

2.1 点云空间坐标转换

在场景数据中，深度图像与点云都包含三维空间信息，两者也可以互相转化。在相机成像模型中包括相机坐标系、像素坐标系、图像坐标系和世界坐标系。深度图像是像素坐标系的数据加深度信息，点云数据是世界坐标系的三维空间数据。深度图转换为点云，实质上是将图像的坐标通过一定变换，转换到图像坐标系，再转换为世界坐标系得到。

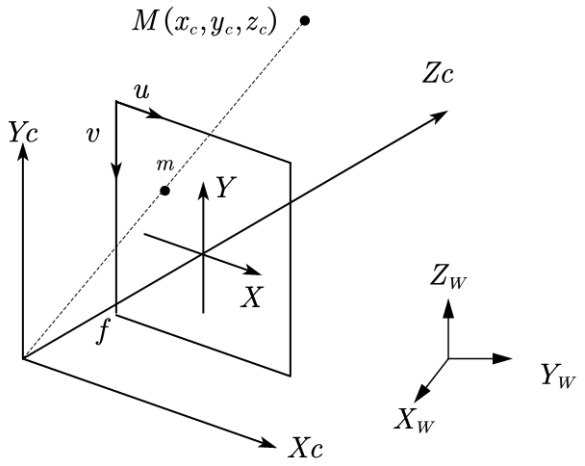

图 2-1 成像示意图[66]

相机成像过程如图 2-1 所示，图中世界坐标系为(X_W, Y_W, Z_W)，相机坐标系为(Xc, Yc, Zc)，图像坐标系为(X, Y)。像素坐标系原点是成像平面的左上角，因此成像点在图像坐标系中的坐标为(u, v)，在世界坐标系中的坐标为(x_w, y_w, z_w)，在图像坐标系中的坐标为(x, y)。世界坐标系中M点转换为图像坐标系中的m点的过程[66]如式(2-1)所示，其中R为相机外参中的旋转矩阵，T为相机外参中的平移矩阵，f为相机焦距。

$$z_c \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{2-1}$$

在图像坐标系中，像素坐标系的原点的坐标为(u_0, v_0)，根据两个坐标系原点之间的位置关系，可以得出图像坐标系的坐标(x, y)与像素坐标系的坐标(u, v)相互转化的公式，如式(2-2)所示，其中d_x和d_y表示在u和v轴上像素的物理尺寸大小。

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} d_x & 0 & 0 & -u_0 d_x \\ 0 & d_y & 0 & -v_0 d_y \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{2-2}$$

在世界坐标系中，原点与相机坐标系一致，坐标轴方向一致，因此R矩阵和T矩阵可分别简化为单位矩阵和零矩阵，并将式(2-1)和式(2-2)结合，可得式(2-3)，即像素坐标系和世界坐标系的转换公式[66]。

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{d_x} & 0 & 0 & u_0 \\ 0 & \frac{1}{d_y} & 0 & v_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{2-3}$$

2.2 点云特征提取

2.2.1 共享多层感知机技术

多层感知机（Multi-layer Perceptron, MLP）是一种人工神经网络，主要由输入层、隐藏层和输出层组成，如图 2-2 所示，其中隐藏层可以有多层。多层感知机中除了输入层，其他每层的节点都包含非线性激活函数，可以更好地拟合复杂函数。

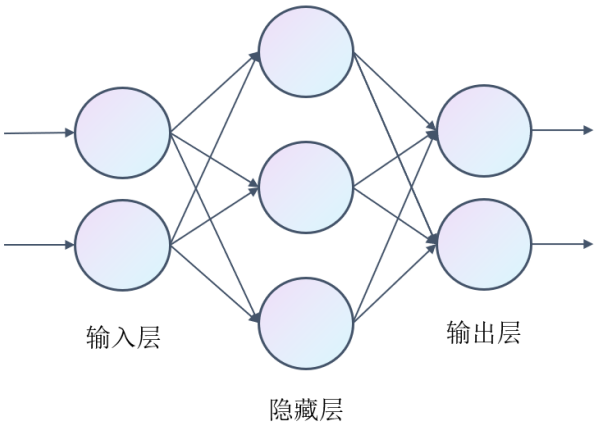

图 2-2 多层感知机

共享多层感知机[33]（Shared Multi-layer Perceptron, Shared-MLP）是 PointNet 处理点云数据时采用的方法，原理与多层感知机一致，其核心思想是将所有点的信息看作整体信息，对点云所有的点应用同样的权重。假定点云输入数据为 N x 3，输出数据为 N x 64，则多层感知机参数为 3 x 64，此时点云数据被看作一个整体，每个点的权重都是多层感知机的参数向量。此外，共享多层感知机也可采用卷积进行实现，设置卷积核大小为 1，步长和填充设置为 0，和上述例子一致，卷积层会生

成 64 x 1 x 3 的卷积核，卷积核沿点云数量方向进行卷积，N 个点的卷积核共享。本文中应用于点云的 MLP 即代表 Shared-MLP。

2.2.2 点云数据特点

点云数据是在欧氏空间内的点的一个集合，具有无序性、转换不变性、稀疏性、密度不均匀等特点。点云数据的组织方式是对顺序不敏感的，无论数据中每个点的顺序如何排列，整体数据所表示的点云不变。同时，点云数据中的数值变换若由空间变换造成，例如旋转或平移，此时数据仍旧没有实质性的变化，即转换不变性。稀疏性表示点云之间是散乱排列，和图像像素不同，并且点与点的距离没有规则，受设备、环境条件的影响，点云的密度在不同区域会发生大幅度变化。

点云数据的组织形式一般为二维矩阵，例如$N \times D$，N代表点云数量，D代表数据维度。点云最基础的数据即三维空间中的坐标信息，在此之外，通道信息还可能包括坐标、颜色、法线等信息。

三维点云学习面临着如下两个挑战：（1）三维点云的无序性，网络需对不同顺序的点集数据可得到相同的输出。（2）三维点云的转换不变性，三维点云因扫描时角度不同，从而得到的点云坐标不同，但这两组坐标不同的点云仍表示同一个物体，因此网络需对这样的点云数据得到相同的输出。

2.2.3 PointNet

作为点云学习网络的先驱，PointNet[33]给出了在深度学习上解决无序性和转换不变性的方案，即采用对称函数和转换网络的方法。

（1）无序性解决方案——对称函数

点云数据通常表示为一个二维矩阵$N \times D$，代表N个点，每个点有D维特征。N个点的顺序可以随意打乱，因此需要一个对称函数对任意顺序的输入数据，始终保持相同的值。

如式(2-4)所示，f表示一个连续集合函数，指代网络结构；x_i表示点云中的某个点。对称函数π对任意输入的n个点进行位置关系置换，例如对$x_1, \dots, x_n$作最大化或求和操作，该函数会始终保持相同值。

$$f(x_1, \dots, x_n) \equiv f\left(x_{\pi_1}, \dots, x_{\pi_n}\right) \tag{2-4}$$

上述公式可以解决无序性的问题，但同时丢失了许多点的信息，仅仅保留了部分信息。因此，对式(2-4)进行改进，得到式(2-5)。

$$f(\{x_1, \dots, x_n\}) \approx g\left(h(x_1), \dots, h(x_n)\right) \tag{2-5}$$

式(2-5)中，h是特征提取函数，可将R^N的特征映射到R^N；g为对称函数，可将$(R^K \times \cdots \times R^K)_{N\times 1}$的特征映射为$R$。通过$h$函数，将点$x_i$的信息映射到高维空间。由于高维空间容易出现冗余的信息，因此在高维特征上进行最大化操作避免了大量特征信息的丢失。

$$f(\{x_1,\ldots,x_n\}) \equiv \gamma \circ g\big(h(x_1),\ldots,h(x_n)\big) \tag{2-6}$$

最后将式(2-5)改写为式(2-6)，只需确保g是对称函数，便可保证整个函数都是对称的，如图 2-3 所示。对称函数g可使用最大或平均池化函数，也可使用求和函数，在 PointNet 中，使用共享多层感知器来描述函数h和γ，g函数采用最大池化。

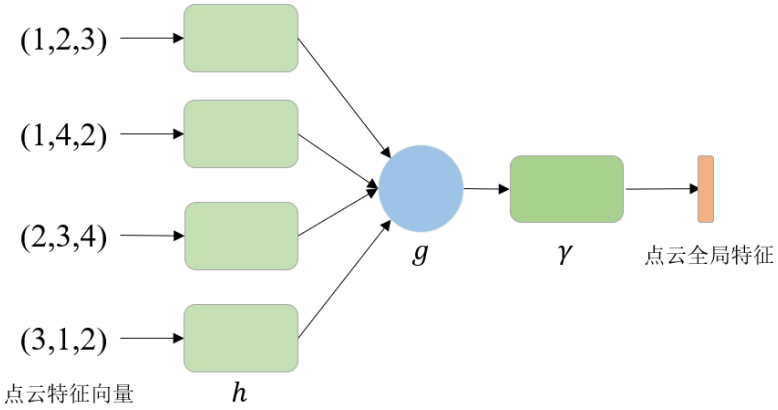

图 2-3 PointNet 无序性解决方案[33]

（2）转换不变性解决方案——T-Net

PointNet 认为可以通过网络学习到一种易于特征学习的点云的数据形式，因此首先通过转换矩阵对原始点云数据进行一定的对其，使得数据更加易于网络学习，转换矩阵由一个小型的神经网络（T-Net）训练得到。

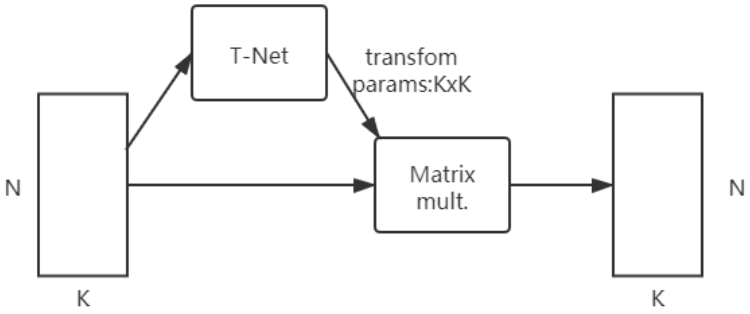

图 2-4 T-Net 网络[33]

如图 2-4 所示，T-Net 的输入为$N \times K$的矩阵，通过训练得到一个$K \times K$的转换矩阵，将$N \times K$的输入点云与变换矩阵进行矩阵相乘，完成数据的变换。此外，三维点云的旋转不变性可以通过约束点的变换为正交变换达到[33]，因此在损失函数中引入正则化项，将转换矩阵的点云信息约束为正交矩阵，如式(2-7)所示，L_{reg}是约束变换矩阵的损失项，I 是对应于输入矩阵维度的单位矩阵，A 即是变换矩阵。

$$L_{\mathrm{reg}} = \|\mathrm{I} - \mathrm{AA}^{\mathrm{T}}\|^2 \tag{2-7}$$

PointNet 网络结构如图 2-5 所示，上方为分类分支，下方为分割分支，网络采用两次 T-Net 转换，对称函数为最大池化函数。在分类分支中，点云通过 T-Net 对数据进行转换，通过共享多层感知机提取特征，最终整体点云的特征被池化到一维特征向量，通过全连接层输出分类分数。在分割分支中，将整体点云的特征向量与每个点的特征向量相拼接，再通过共享多层感知机学习到每个点的预测分数。

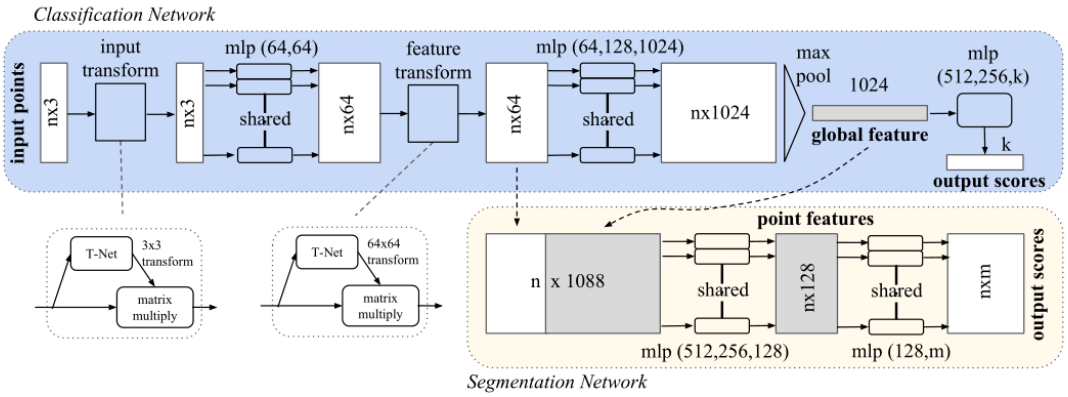

图 2-5 PointNet 网络结构图[33]

2.2.4 PointNet++

在 PointNet 的基础上，PointNet++[34]引入了不同粒度的特征提取思想，提出了层次化的 PointNet，可以更好地学习到点云的局部特征。层次化结构的主要流程是采样、分组和特征提取。

如图 2-6 所示，网络结构分为层次化点云特征学习（Hierarchical point set feature learning）和具体任务的解码器。在层次化点云特征学习模块中包括两个点云特征抽象模块（Set abstraction），该模块包括点云采样、分组和 PointNet 特征提取。PointNet++通过两个特征抽象模块对点云进行下采样，逐步将特征提取的范围缩小，得能表示点云局部特征的子集，并对子集进行特征提取。在解码器部分，分类解码器通过多个线性层得到分类结果。分割解码器中的特征传播模块采用特征插值算

法将子集特征插入整体点云特征中，堆叠多层特征传播模块得到输入点云的逐点语义特征，并通过共享多层感知机学习出每个点的语义预测。

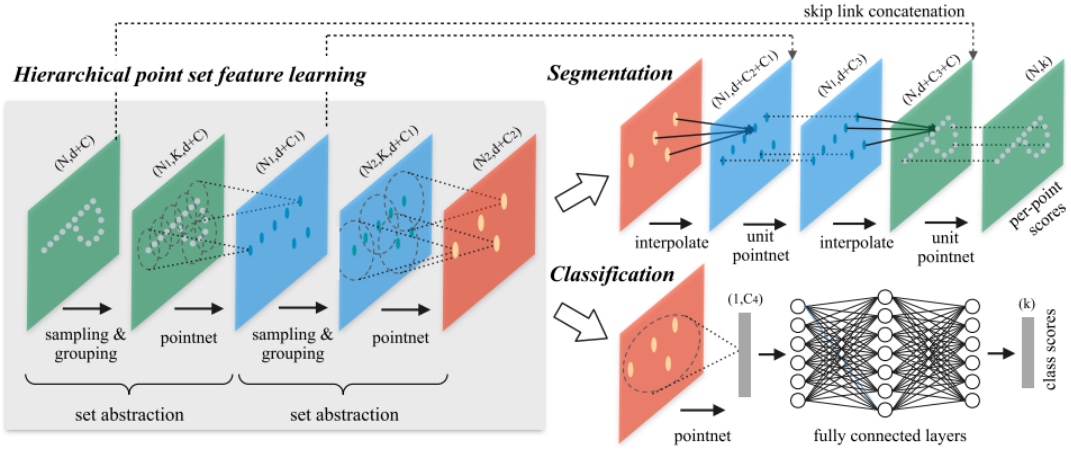

图 2-6 PointNet++网络结构图[34]

2.3 注意力机制

注意力机制作为近些年广受关注的技术，在深度学习的许多领域不断取得了新的成绩。注意力机制主要通过计算注意力分数，模拟人类在理解外界信息时的选择性增强和抑制的行为，把握主要信息，降低无效信息的影响，从而达到增强对数据的理解能力。注意力机制一般由查询（Query）、键（Key）和值（Value）组成，首先计算 Query 和 Key 的相似性，一般由 Attention 矩阵表示，然后将相似性与 Value 进行加权计算，例如矩阵相乘，如此便得到了 Value 的加权结果，即应用注意力机制后的特征，如上过程可以用式(2-8)表示，其中S代表相似性函数。

$$Attention(Query, Key, Value) = S(Query, Key) \cdot Value \tag{2-8}$$

对于相似性的计算，常见的方法[67]有向量点积、余弦相似度、MLP 网络学习等。对于注意力机制，最简单相似度计算方式是直接通过 MLP 网络学习 Value 的自相似性，并且 Key 和 Value 是同一个，如式(2-9)所示。

$$Attention(Query, Key) = MLP(Query) \cdot Key \tag{2-9}$$

相似性点积模型通过计算 Query 和 Key 的矩阵点积，并对其结果进行归一化，得到注意力分数，如式(2-10)所示，其中d为 Query 和 Key 的特征维度。

$$S(Query, Key) = (Query \cdot Key)/\sqrt{d} \tag{2-10}$$

Transformer[55]是谷歌在 2017 年提出的 Seq2Seq 模型，在机器翻译等领域表现出优异的成绩。自注意力（Self-Attention）和非串行化是 Transformer 的特点，非

串行化是指在训练时无需数据按照一定顺序输入，而是同时训练，并行处理，与之相反的，RNN[68]与 LSTM[69]具有串行的特点，训练时需要处理完当前数据才可以处理下一个数据。

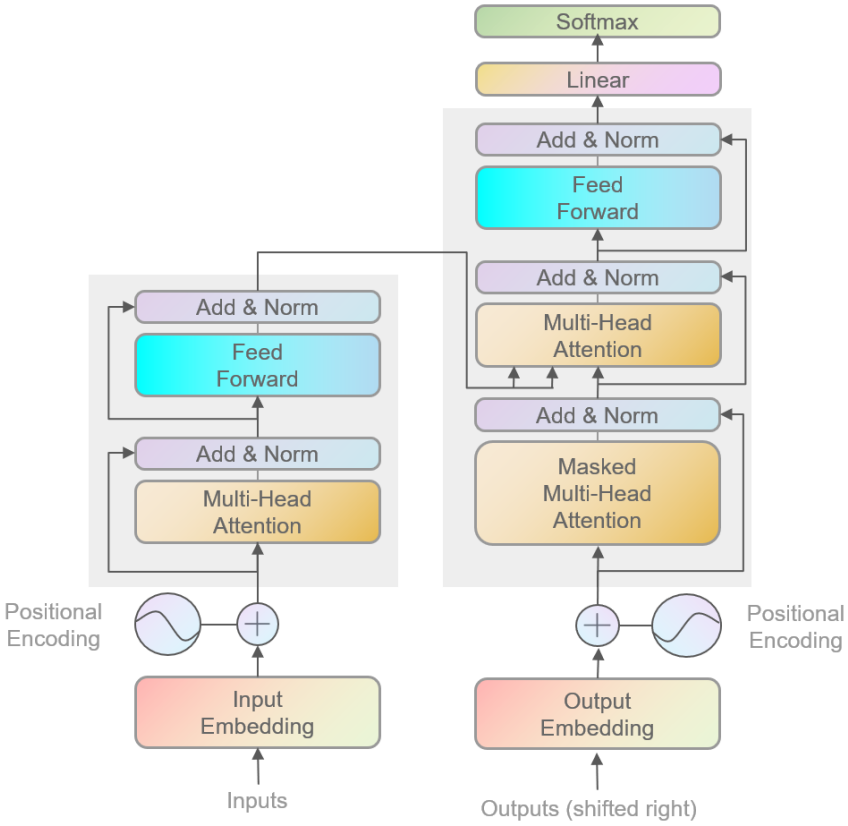

图 2-7 Transformer 模型结构[55]

Transformer 结构如图 2-7 所示，左侧分支为编码器，右侧分支为解码器。编码器将输入的序列编码为高维特征，解码器将高维特征解码为新的序列。在机器翻译任务中，输入为语言序列，输出则是翻译后的语言序列。在编码器中，首先输入的语言序列经过输入嵌入（Input Embedding）被编码为词向量，在位置编码（Position Encoding）中将词的位置信息嵌入，随后，多头注意力模块（Multi-Head Attention）通过多个自注意力模块对词向量进行并行的特征编码，最后通过前馈神经网络（FeedForward）得到高维特征。在多头注意力模块和前馈神经网络后都有残差结构和特征的正则化操作。

Transformer 的核心是自注意力模块，自注意力模块采用点积计算相似性，计算方式和注意力机制相似，但自注意力机制在计算相似性时采用并行化计算，例如对某个语言序列进行翻译时，所有词的特征通过网络生成 Query、Key 和 Value，注意力机制会对每个词单独计算注意力分数，而自注意力机制计算每个词和其他

词的注意力分数，更好地学习到整句话内部的关联关系，Transformer 中自注意力模块结构如图 2-8 所示。

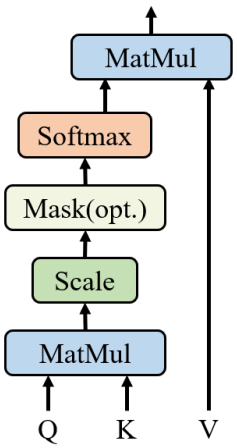

图 2-8 Transformer 中的自注意力结构[55]

2.4 本章小结

本章介绍了与本文研究相关的一些基础方法和理论知识，主要从三个方面对点云相关基础进行了说明，分别是点云空间坐标转换、点云特征提取和注意力机制，并对上述理论进行了阐述和分析，介绍了图像像素与三维点云之间的坐标转换方法，随后分析了点云语义提取存在的问题和挑战，对点云特征提取进行了系统的说明，最后对注意力机制进行了简要的讲解。近年来点云语义分割算法开始了高速发展，分割精度不断提高，但仍然存在一些待解决的问题，后续章节的研究将以上述技术为基础，针对点云分割领域中存在问题，对点云语义分割算法进行深入的分析和相应的创新。

第三章 融合 RGB 特征的点云语义分割网络

3.1 引言

场景数据包括以二维图像为代表的二维数据和以 RGB-D 深度图、三维数据为代表的三维数据。二维图像受光照条件、复杂环境等影响，成像质量会有一定下降，同时物体遮挡也是二维图像不能更好描述现实世界的重要因素。RGB-D 深度图是在二维图像的基础上，增加了像素的深度信息。三维点云是一种稀疏的和非结构化的三维数据，用来描述现实世界中物体形状和位置。

语义分割作为场景理解的基础技术[17]，已经成为当下计算机视觉研究的热点。二维图像与三维点云，两者都包含一定的结构信息和语义信息。相比于二维图像，三维点云在表示现实场景上具有更多的细节和特征，具有更加丰富的结构信息和语义信息。随着激光传感器和深度传感器的快速发展和应用，点云的获取难度降低，三维扫描技术的发展也促进了三维数据的增长和质量的提高，网络上涌现出许多完善的三维点云场景数据。

计算机二维视觉中，最流行的当属卷积神经网络。卷积神经网络的出现使得计算机视觉任务得到了前所未有的发展，传统的手工特征描述算子无法比拟其强大的特征提取能力。卷积神经网络由卷积层、池化层、全连接层组成，卷积操作的输入是规格化的网格，如 RGB 图像。卷积层通过许多卷积核提取不同感受野上的特征，池化层进一步筛选特征，全连接层将特征转换为预测分数。

PointNet 是点云分割任务中基于点的方法的开山之作，随后基于点的方法被频繁研究，而在 PointNet 提出之前，受计算机二维视觉发展所影响，研究者期望将点云转换为卷积神经网络可以处理的数据。因此点云被转换为多视图、体素等规格化的数据。然而，多视图数据丢弃了三维结构的空间数据，体素方法则需要大量的计算资源。

大量的二维数据与三维数据可以更好地描述场景信息，也包含了更多的语义信息。当两种数据可配准时，网络可以充分利用上述数据进行视觉任务的实现。针对上述问题和对场景数据的精准分割，本章设计了一种特征融合方法。将多视图数据与点云数据相结合，在此基础上提出了一种融合 RGB 图像特征的三维点云分割网络，包括提取 RGB 特征的二维网络、点云数据与 RGB 特征的融合模块、分割融合数据的点云网络。该网络可以充分利用场景数据进行计算机视觉任务，提高网络的特征提取能力。通过对比实验和消融实验验证了网络和各模块的优越性。

3.2 算法工作

本节详细介绍融合 RGB 特征的点云语义分割网络（Joint RGB-feature Attention Net, JANet），首先描述网络整体结构，随后介绍各个模块并分析原理。网络在对应场景点云的视图上提取图片的二维语义特征，并将其映射到三维空间中。设计特征融合算法将二维语义特征融合到三维点云数据中，融合后的三维数据输入三维点云分割网络中进行训练。

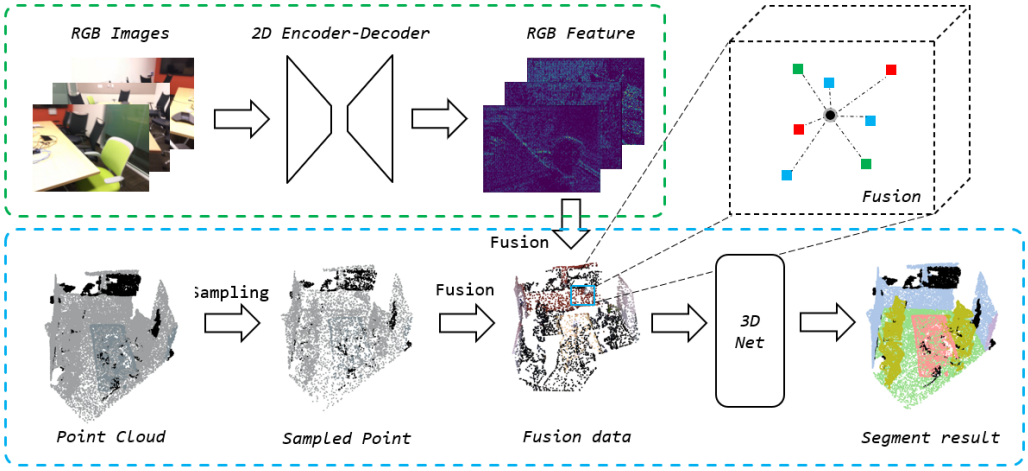

图 3-1 JANet 网络整体架构

如图 3-1 所示，JANet 总体结构分为上方的 RGB 特征提取和下方的三维点云分割两个分支，两个分支由特征融合模块连接。在 RGB 特征提取分支中，采用二维分割网络来得到场景视图的二维特征，该网络将场景的视图图像编码为高维度特征图像。随后将该特征图像输入特征融合模块，通过注意力机制将特征图像和三维点云数据进行融合，最终输出增强的融合点云数据。通过将融合数据输入三维点云分割网络，获得分割结果。

在三维点云分割部分中，为了捕捉局部点云特征，设计了相对特征提取（Relative Feature Extraction, RFE）模块来捕捉三维空间中点与邻域的特征关系，更具体地说，使用位置编码（Relative Position Encoder, RPE）来编码邻近点与中心点的关系，并通过神经网络对这种关系的特征进行提取。为了更好地在融合数据上进行空间特征提取，设计了交叉空间注意力（Cross Spatial Attention, CSA）模块，该模块和特征融合模块都可以对特征进行语义修正，达到辅助特征提取的目的。本章提出的算法的计算流程如算法 3-1 所示。

算法 3-1 融合 RGB 特征的点云分割算法

算法 1：融合 RGB 特征的点云分割算法
输入：图片信息 B x 3 x H x W，深度信息 B x 1 x H x W，点云信息 B x N x 3
参数：迭代次数 epoch，学习率 lr，类别数量 num_class，视图数量 num_view，采样点数 num_points，邻近点数 k
输出：B x num_class
算法流程：
1. 预处理数据集，根据点云与图片的位置关系选取可融合的视图图片
2. for t←1,2…T, do
3. 在场景二维图片上训练二维分割网络
4. End for
5. for t←1,2…T, do
6. 输入图片信息 B x 3 x H x W，二维网络获取图片特征 B x C x H x W
7. 输入深度信息 B x 1 x H x W，点云信息 B x N x 3，通过深度信息将图片特征逐像素投影到三维空间，根据公式将图片特征融合到点云信息中，得到融合点云数据 B x N x D
8. 在融合点云信息上训练三维语义分割网络，输出每个点的预测分数
9. End for

3.2.1 场景图像分割及特征融合

图像的特征信息可以通过传统二维分割网络得到，本章算法采用 U-Net[70]主干作为二维网络，在进行点云分割前完成二维网络的训练，使得网络达到一定的准确率。训练好的二维网络作为三维网络训练中的二维特征编码器来提取图像的特征，不参与三维点云网络的训练，下一步在特征融合模块中将图像特征与点云数据相融合。此外，在 U-Net 中引入 ResNet 结构，增强对图像的特征学习能力。其中 U-Net 相比于传统卷积神经网络，区别在于跳层连接和 U 型结构。上述两种结构可以让网络提取到高分辨率和低分辨率的特征，并且对于不同尺寸和特征维度的数据，其上下文信息和位置信息也被网络较好地捕捉到。ResNet 通过残差结构将深层网络和浅层网络的特征对应成恒等映射，解决了网络在层数过深时的退化问题，使得神经网络可以尽可能堆叠。

针对点云和图像的融合问题，受 MVPNet[51]网络的启发，为将两种数据更有效地融合，本文设计一种基于距离的特征融合方法，如图 3-2 所示，其中彩色正方形表示像素形式的二维图片高维特征，黑色圆形代表点云中某个点，通过图像深度信息，将二维像素映射到三维空间，通过全局点云位置信息和图片像素位置信息进行特征融合，达到补充点云数据的语义信息的效果。

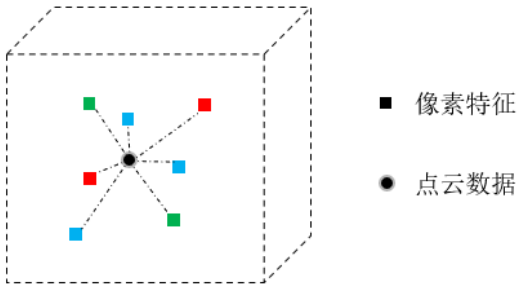

图 3-2 融合算法示意图

在特征融合模块中，假设输入 N 个 H x W x C 的二维特征图和一组点云。首先将二维特征图的像素点投影到三维空间中，并对密集的像素进行下采样，得到 M x C 的像素点云，像素点云中每个点由像素坐标及深度信息计算而来，其中 M < N x H x W。

如式(3-1)所示，对于点云中的点p_i，使用欧几里德距离在三维空间内找到其最近的K个像素点云的点$\{v_i^1 \dots v_i^k \dots v_i^K\}$，其中$v_i^k$代表 M x C 的像素点云的点的空间坐标，$\oplus$代表拼接操作。

$$D_i^k = p_i \oplus v_i^k \oplus (p_i - v_i^k) \oplus \|p_i - v_i^k\| \tag{3-1}$$

通过编码像素点与点云的位置信息D_i^k，模拟计算每个像素点v_i^k对p_i的语义贡献。最终得到一个邻近像素点编码的集合$D_i = \{D_i^1 \dots D_i^k \dots D_i^K\}$，并在下一步中和图像特征进行融合。

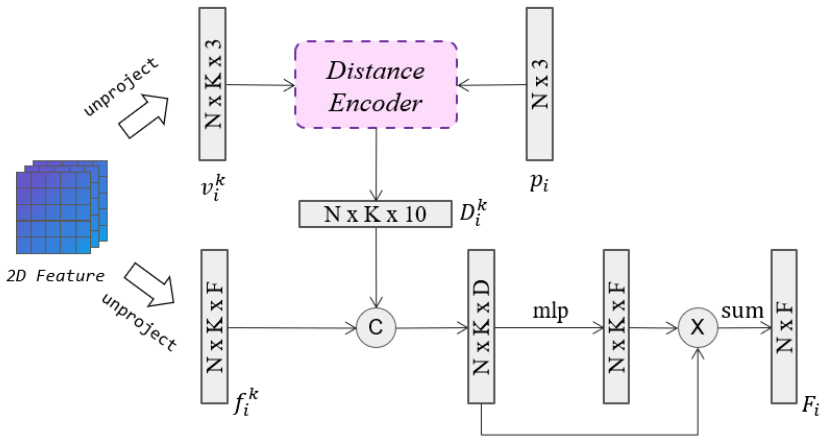

图 3-3 融合模块结构

$$F_i = \sum_k \left(\left(f_i^k \oplus D_i^k \right) \odot MLP\left(f_i^k \oplus D_i^k \right) \right) \tag{3-2}$$

如图 3-3 所示，特征融合模块将每个点的信息集成了邻近像素点的特征，其中F_i代表融合后点的信息，融合了点p_i和其邻近像素的特征信息f_i^k，并通过f_i^k增强融合数据的语义信息，计算过程如式(3-2)所示。

3.2.2 点云相对特征编码

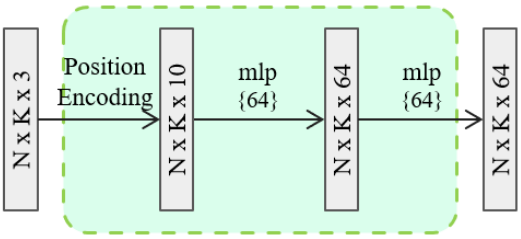

图 3-4 相对位置编码

如图 3-4 所示，相对位置编码（Relative Position Encoding, RPE）模块用来编码关键点和邻近点的位置信息。给定点云P和每点特征，对中心点p_i的最近的k个点$\{p_i^1, p_i^2, \cdots, p_i^k\}$的进行位置信息编码，并设计一个共享函数$g$来学习每个编码信息的特征。$g$函数包含一个或多个 SharedMLP，图 3-4 中 RPE 模块包含两个输出维度为 64 的 SharedMLP，并且其中包括 ReLU 激活函数进行非线性学习。RPE 模块公式如式(3-3)所示。

$$\hat{p}_i^k = g\left(p_i \oplus p_i^k \oplus \left(p_i - p_i^k\right) \oplus \left\|p_i - p_i^k\right\|\right) \tag{3-3}$$

其中p_i和p_i^k分别表示中心点和其邻近点的空间位置，$\oplus$表示拼接运算符，“$\|\ \|$”符号计算欧几里得空间的距离。$\hat{p}_i^k$编码了局部邻域的点的相对位置信息，并通过g函数将这种相对位置信息抽象为高维特征。

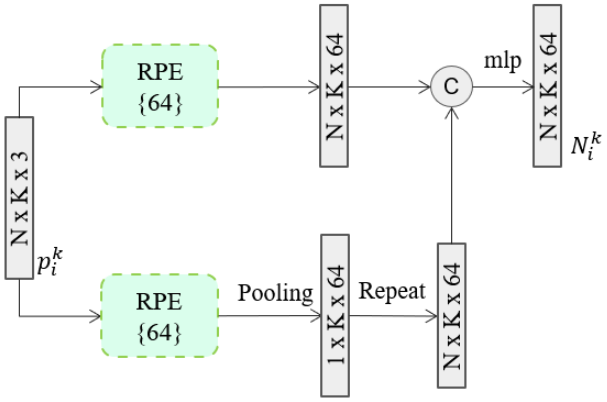

图 3-5 相对特征编码

如图 3-5 所示，相对特征提取（RFE）模块由两个相对位置编码（RPE）模块组成，旨在捕捉关键点和邻近点的空间分布信息。在相对特征提取模块中，对每个点 p_i^k，通过两个相对位置编码模块分别编码邻域内的局部特征和全局特征，并拼接为邻域空间特征N_i^k，公式如式(3-4)所示，其中W_0和W_1分别是两层 MLP 的权重矩阵。

$$N_i^k = W_0 p_i^k \oplus \left(\frac{1}{K}\sum_{i=1}^{K} W_1 p_i^k\right) \tag{3-4}$$

3.2.3 点云交叉空间注意力

在特征抽象阶段，共享多层感知机（Shared-MLP）模块作为一种有效的编码器，结合注意力机制，可以较好地完成特征提取任务。交叉空间注意力(CSA)模块如图 3-6 所示，核心思想是通过多层感知机处理邻域空间特征，交替生成注意力矩阵和下一层局部特征并相乘，达到空间特征提取和语义信息增强的效果。

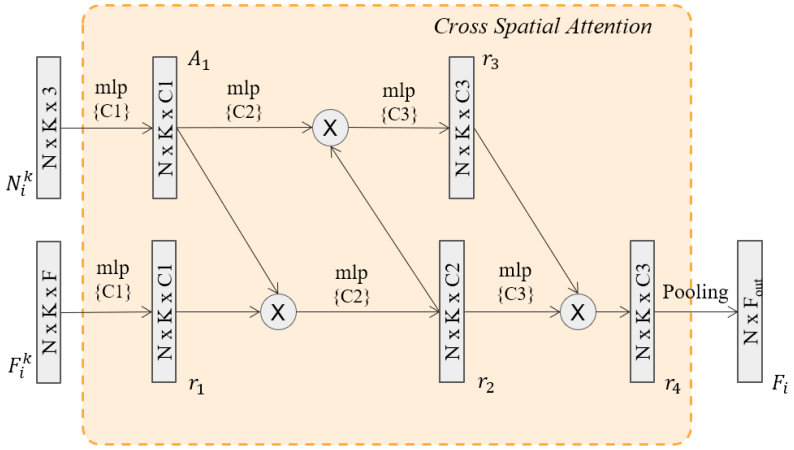

图 3-6 交叉空间注意力

设计了一个g函数来生成具体的注意力矩阵和局部特征，函数g由一组共享多层感知机和 Softmax 组成，注意力矩阵A_l定义如式(3-5)所示。其中A_l表示当前l层下的注意力矩阵，W是共享多层感知机学习到的参数矩阵，r_l代表学习过程中的中间特征，如式(3-6)所示。

$$A_l = \begin{cases} g(N, W) & , i = 1 \\ g(A_{l-1}, W) & , i = 2, 4, \ldots \\ g(r_{l-1}, W) & , i = 3, 5, \ldots \end{cases} \tag{3-5}$$

$$r_l = g(A_{l-1} \cdot r_{l-1}) \tag{3-6}$$

如图 3-6 所示，在交叉空间注意力模块中进行了 3 次矩阵元素相乘。对于给定点云邻域空间特征N_i^k和局部特征F_i^k，CSA 模块学习聚合局部特征，并通过语义增强来修正特征，最终由最大池化得到编码后的特征信息向量F_i。

3.2.4 融合 RGB 特征的点云分割网络

三维点云分割网络 JANet 结构如图 3-7 所示，该网络主干为 PointNet++，其中编码器由多层相对特征编码和交叉空间注意力组成。该网络结构可以较好地捕捉不同粒度的点云特征，进而预测融合点云数据的语义标签。

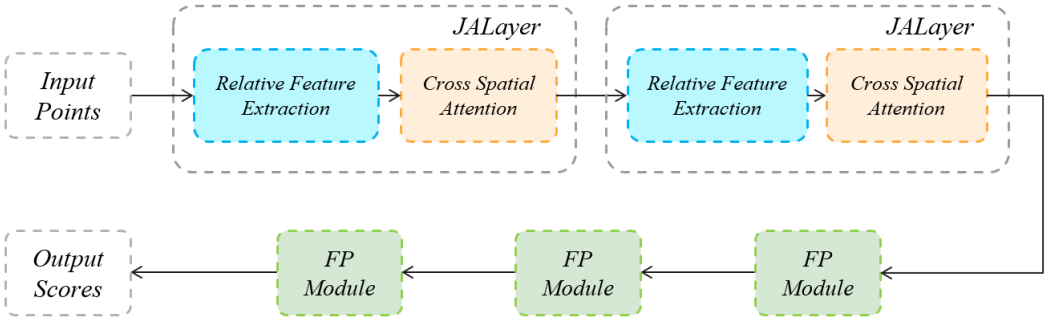

图 3-7 三维点云网络结构

点云数据是无序的点的信息，通过共享多层感知机将其抽象为特征后，和二维卷积不同，每个点的感受野并没有因此而扩大，因此，针对上一步提到的融合数据，设计一种联合注意力层（Joint Attention Layer, JALayer）来更好地对点云的空间特征和语义特征进行提取。在三维点云分割网络中，采用多层 JALayer 对不同的中心点进行特征提取。每层 JALayer 由一层相对特征提取和一层交叉空间注意力组成，详细结构如图 3-8 所示。

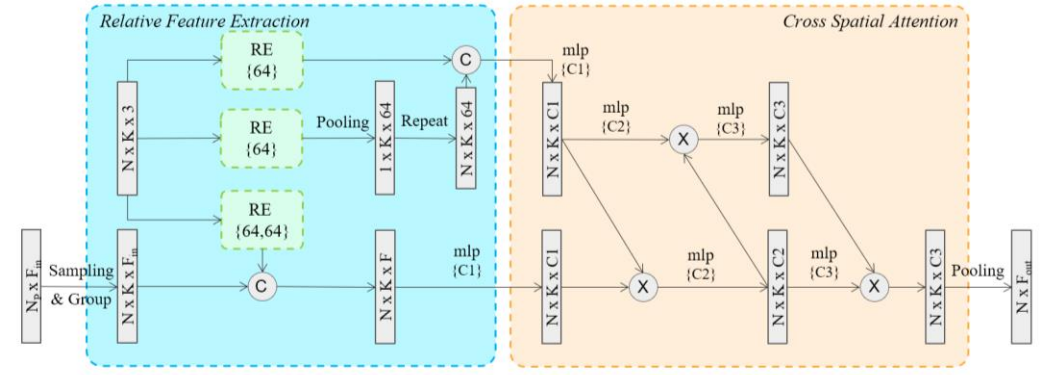

图 3-8 JALayer 结构图

在实际应用中，一个场景所包含的点云数量庞大，即便是单个物体点云也会由非常多的点来组成。传统的卷积神经网络在进行特征提取和预测时，要求被处理数据是结构化的，并且神经网络的参数随着数据量的增加而增加。对于点云数据中海量的点，通过采样和分组的方式，从点云中提取可代表点云的关键点，以及关键点附近的邻近点。通过这种方式减小神经网络负担，提高训练效率。在 JALayer 中，点云首先进行最远点采样，采集出可代表该点云的关键点，并通过球查询得到每个关键点的邻近点，随后关键点和邻近点的坐标信息和特征信息送入相对特征编码模块进行空间特征编码。

3.3 算法结果与分析

本节根据本章研究的方法设计了一个融合 RGB 特征的点云分割网络，为了验证该网络的有效性及综合性能，在 ScanNet[7]数据集上设计了多组实验来验证网络在分割任务上的效果。实验基础环境为：Ubuntu 18.04，RTX 2080Ti，Pytorch 1.2.0，详细环境如表 3-1 所示。

表 3-1 实验环境

系统平台	CPU	GPU	显存	RAM	Pytorch	Cuda
Ubuntu 18.04	i7-8700	RTX 2080Ti	11GB	16GB	1.2.0	10.1

网络效果评估需要量化的指标，分类和分割常见的评价指标有精确度（Accuracy）、交并比（Intersection over Union, IoU）、召回率（Recall）等。交并比（IoU）是语义分割任务中广泛使用的指标，本研究采用每个类别平均的交并比（mIoU）作为网络的评价指标。

通常来说，在二分类数据集中，数据集中的数据分为正样本（Positive）和负样本（Negative），多分类数据集中，数据则为多个类别。网络对输入数据进行预测，得到预测的标签，预测标签和数据真实标签相同的记为 True，预测标签和数据真实标签不同的记为 False。样本为正样本并预测为正记为 TP（True Positive），样本为负样本并预测为负记为 TN（True Negative），样本为负样本并预测为正记为 FP（False Positive），样本为正样本并预测为负记为 FN（False Negative）。

在分割任务中，交并比（IoU）表示预测正确的部分和实际正确部分的相似度，具体如图 3-9 所示，其中红色部分为 FN，代表预测为负但实际为正样本的集合，绿色部分为 FP，代表预测为正但实际为负样本的集合，上述两个部分都是预测错误的部分，中间黄色部分是 TP，代表预测为正并且实际也为正的样本，是预测正确的部分。

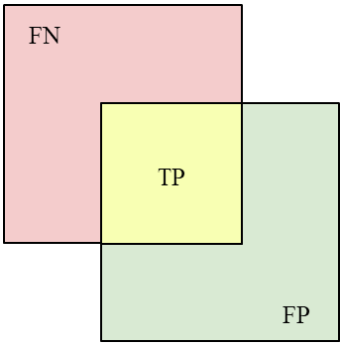

图 3-9 IoU 示意图

根据图 3-9，可以得到 IoU 的计算公式，如式(3-7)所示。

$$IoU = \frac{TP}{TP+FP+FN} \tag{3-7}$$

在点云分割任务中，mIoU 计算每个类别或者场景的 IoU 的平均值，计算公式如式(3-8)所示。其中 C 代表类别数量或者场景数量。

$$mIoU = \frac{1}{C}\sum_{i=1}^{C}\frac{TP}{TP+FP+FN} \tag{3-8}$$

3.3.1 数据预处理

ScanNet[71]是一个室内场景数据集，该数据集使用 iPad 的内部摄像头和一个额外的深度摄像头拍摄。该数据集每次扫描的数据由一个带有相关姿势的 RGB-D 序列、一个完整的场景网格以及语义和实例标签组成，共有 2.5M 视图图像和 1513 个点云网格，包括 21 个语义类别的对象。数据集中每个场景都包括二维数据和三维数据，二维数据是场景采集的视频帧，三维数据是点云信息。本算法使用 1201 个扫描数据用于训练，312 个扫描数据用于测试。图 3-10 表示 ScanNet 数据集中一个 3D 场景的真实图像和标注点云。

图 3-10 ScanNet 数据集(a)示例图片与(b)标注点云

为了提高网络训练效率，节省网络训练时间，在预处理步骤计算场景图像与点云的关联关系。首先对场景点云进行下采样，选取场景点云的关键点，通过图像的

深度信息将图像像素投影到三维空间，计算像素与点云的空间距离，设置一个距离阈值，当像素与该关键点的距离小于该阈值，认为当前像素可以代表该关键点，图像中该类像素越多，认为此图像代表场景点云的能力越强。对每个场景计算场景点云和场景图像的关系，并选取十张代表能力最强的场景图像作为后续训练数据，将图片编号和场景点云共同打包，得到网络可使用的训练数据和测试数据。

3.3.2 实验设置

在本章网络中，二维图像网络采用 U-Net 结构，编码器为预训练的 ResNet 网络。三维点云分割网络采用本章提出的 JANet，在训练阶段，在预处理阶段保存的图像中选取 3 张场景图像，融合算法的邻近像素数量设置 3。网络训练的详细参数如表 3-2 所示。

表 3-2 实验参数

Model	Iteration	BatchSize	学习率	NumViews	NumPixels	优化器
2D Net	80000	32	0.005	-	-	SGD
3D Net	40000	16	0.004	3	3	Adam

在本章实验中，由于设备显存限制，网络训练时视图数量最大为 3。为使网络学习足够的二维语义信息，将训练视图设为 3，对验证时视图数量进行对比实验，如表 3-3 所示。验证网络效果时，视图数量为 5 的分割精度最高，达到了 68.2%，相比于视图数量为 3 和 1 分别高出 0.6%和 3.7%，同时高于视图数量为 7 时的精度。当视图数量增加时，二维网络的错误预测会影响融合算法对数据的语义补充，因此选取合适的视图数量可以增强网络的学习能力。本章网络在训练时视图数量设置为 3，验证时视图数量设置为 5。

表 3-3 视图数量实验

Method	NumViews	mIoU
JANet	1	64.5
JANet	3	67.6
JANet	5	68.2
JANet	7	68.1

3.3.3 对比分析

在 ScanNet 测试集上进行实验评估，评估指标是 20 个类别的平均 IoU (mIoU)。JANet 在 ScanNet 数据集上的 mIoU 表现为 68.2%，均高于其他优秀算法。本章采

用的 U-Net 网络在 ScanNet 上的分割结果为 61.1%，PointNet++网络的分割结果为 54.5%。与上述两种网络相比，JANet 的分割效果有明显提升，分别提升了 7.1%和 13.7%，表明 JANet 同时集成了二维图形和三维形状的分割能力，也证实了将 RGB 图像特征提升到 3D 进行语义补充的有效性。此外，JANet 较 MVPNet 在 mIoU 上有 0.3%的提升，证明了本章算法和各模块的优越性。

表 3-4 ScanNet 分割结果对比

Method	U-Net[70]	PointNet++[34]	PointConv[72]	MVPNet[51]	Ours
mIoU	61.1	54.8	58.0	67.9	**68.2**
bath	58.4	71.9	75.0	78.2	**81.3**
bed	47.8	71.4	67.2	77.0	**77.4**
bkshf	45.8	71.1	71.3	**79.4**	79.2
cab	57.2	46.9	47.4	59.1	**59.1**
chair	36.0	83.3	81.3	87.2	**87.2**
cntr	25.0	57.0	56.8	58.2	**59.7**
curt	24.7	24.4	55.1	**68.0**	67.7
desk	27.8	50.4	52.5	**63.8**	63.0
door	26.1	35.7	34.6	**61.7**	61.6
floor	84.8	94.0	**94.8**	94.4	94.7
other	18.3	36.9	38.7	55.2	**56.8**
pic	11.7	8.2	6.7	**32.3**	32.1
fridge	21.2	31.4	37.0	57.7	**57.9**
shower	14.5	26.0	**57.5**	51.8	52.6
sink	36.4	60.3	59.0	61.6	**61.6**
sofa	34.6	**71.7**	63.3	70.6	69.4
table	23.2	65.3	65.1	72.1	**72.4**
toilet	54.8	81.7	84.0	**90.4**	90.2
wall	77.9	73.0	74.1	80.1	**81.5**
window	25.2	36.1	44.6	**58.8**	58.2

本章网络的分割可视化效果如图 3-11 所示，第一列代表数据集点云 Ground Truth，图中黑色表示数据无语义标签，第二列代表本章网络的可视化分割结果。从图 3-11 可看出，本章网络的效果与真实标签没有明显差别，尽管网络在语义信息上进行了补充，尤其是具有平面形状的物体的语义，网络对距离相近的平面物体

的预测能力仍然需要进一步加强，例如图 3-11 第二行，地板有一部分被错误预测为窗户。

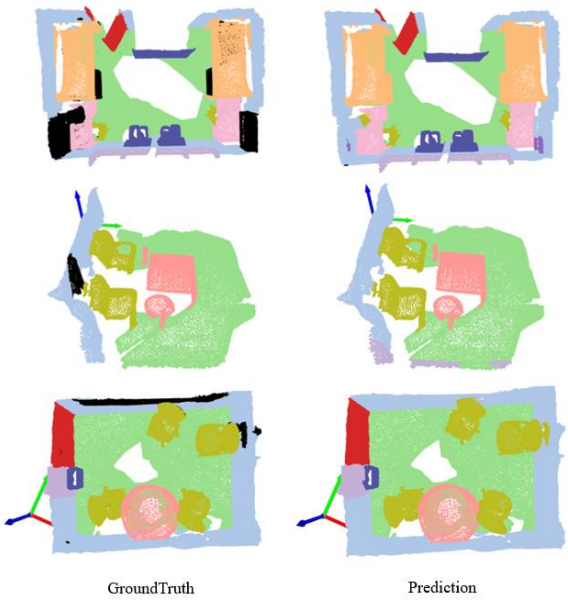

图 3-11 分割结果可视化

3.3.4 消融实验

消融研究是证明三维网络贡献的基本测试方法，目的是验证所提出的算法是否具有真正的适用性。在本节中，将 JANet 的主要部分进行拆分，并将其分别嵌入到网络主干中，来研究每个部分对分割结果的贡献。JANet 包含三个突出的贡献，即相对特征模块、交叉空间注意力模块和融合算法，定量评估结果如表 3-5 所示。

表 3-5 ScanNet 消融实验

方法	融合算法	相对特征编码	交叉空间注意力	mIoU
U-Net[70]				61.1
PointNet++ [34]				53.5
PointConv[72]				58.0
MVPNet[51]				67.9
JANet	√		√	67.9
JANet	√	√		68.0
JANet	√	√	√	68.2

当分别单独使用 RFE 模块和 CSA 模块时，比上述使用本文提出的融合算法的 PointNet 分别高了 0.5%和 0.6%。所有模块组合在一起，以 68.2%的性能展示了模

型的最佳状态，并且优于 PointNet++和 PointConv 等方法。可以看到在模型一定的情况下，丰富的 RGB 语义信息可以提高网络的表现，RGB 特征对点云数据的语义补充是有效的，对分割结果的贡献是显著的。

3.4 本章小结

本文在 PointNet++的基础上，提出了一种融合 RGB 特征的点云分割网络，该网络将二维图像特征投影到 3D 空间中，利用图像像素和 3D 点云的位置信息进行特征融合，实现了对 3D 数据的语义补充。此外，相比于 PointNet++的 Set Abstraction，本章提出的一种新的基于注意力机制的特征提取层 JALayer，更加能够捕捉融合数据的特征。网络在 ScanNet 数据集上进行了评估实验和消融实验，证明了融合 RGB 特征来补充语义的优势，也验证了本章所提出模块的有效性。

第四章 基于 Transformer 的点云学习网络

4.1 引言

点云是一种无序的数据，计算机中存储的点云数据一般为多个点的向量，点与点之间的顺序对点云数据整体没有影响。点云的特征提取或者说特征学习一直是计算机三维视觉中研究的重点。当下方法主要聚焦于点云的局部特征和全局特征，例如 PointNet 通过对称函数提取点云的全局特征，PointNet++对点云采样并分组，将点云分为不同大小的局部区域，进而提取出不同的局部特征。然而，点云的局部特征可以等价为局部区域的全局特征，那么除了点云的全局特征和局部特征，还有一种点与点之间的关系特征并没有被很好地学习到。

自然语言是一种具有顺序的数据，语句之间、字词之间都具有一定的顺序。针对此类序列问题，研究者提出了 LSTM、RNN 等模型处理序列数据，在训练过程中借助顺序信息来提取特征。Transformer 是最近几年研究者针对自然语言处理而提出的方法，其通过注意力机制学习语言序列中词于词之间的关联性，在机器翻译等任务中取得了优秀的成果。不仅仅是自然语言处理领域，Transformer 在计算机视觉领域同样表现出优秀的效果，其中 ViT 通过 Transformer 机制对图像进行分类。

自然语言序列一般由多个句子组成，每个句子由数量不等的词和标点符号组成，在 Transformer 中，输入序列的每个句子被编码为等长的向量，随后进行特征提取。而点云数据的格式可以直接变换为等长向量。近几年，Transformer 在点云学习领域逐渐受到关注，涌现了 Point Transformer[61]和 Point Cloud Transformer[62]等优秀网络，该类网络采用 Transformer 结构对点云数据进行特征学习，在一定程度上对点云特征的内关联性进行了学习。然而点云数据中点与点的关系还存在进一步学习的空间，针对此问题，本章将 Transformer 结构应用到点云学习网络中，提出一种基于 Transformer 的点云学习网络（TransformerNet）。该网络通过特征位置编码（Feature Position Encoding, FPE）将点与点的关系进行编码，并设计点云适应化的 Transformer 编码器对关联特征进行学习和预测。本章的主要贡献如下：

(1) 将自注意力机制应用于点云处理，设计了基于自注意力机制的点云分类网络（Simple TransformerNet），将点云数据作为无序的文本数据经注意力模块进行语义抽象和关联性特征学习。

(2) 捕捉点云中点与点之间的联系，优化自注意力机制，对比分析正则化方法，并设计一种联结注意力模块，更好地提取点云特征内部的关联关系。

(3) 在点云特征上应用特征距离编码，将点云特征表征为关联的边特征，增强网络对点云特征之间的关联关系的学习能力，并设计了一种基于 TransformerNet 的点云学习网络，在点云形状分类、部分分割、场景语义分割任务上进行实验和评估。

4.2 算法工作

在本节中，将详细介绍基于 Transformer 的点云学习网络（TransformerNet）。根据 Transformer 结构，本节首先设计出 Simple TransformerNet，该网络为点云分类网络，采用基本的自注意力机制和 Transformer 编码器结构，并作为本章网络在分类任务上的 baseline。随后在此分类网络上进行优化，设计了特征位置编码和网络正则化方法，构建了基于 Transformer 的点云学习网络的整体结构。

4.2.1 自注意力机制与点云分类网络

Transformer 的结构分为编码器和解码器，其中解码器对输入序列按顺序进行解码，由于点云数据是无序的，因此解码器采用经典的点云网络解码器，不采用 Transformer 解码器。Transformer 的核心在于自注意力机制，本节首先将自注意力机制应用在点云上，通过自注意力机制尽可能地提取点云中点与点的关系，搭建了基于自注意力机制的点云分类网络（Simple TransformerNet）。

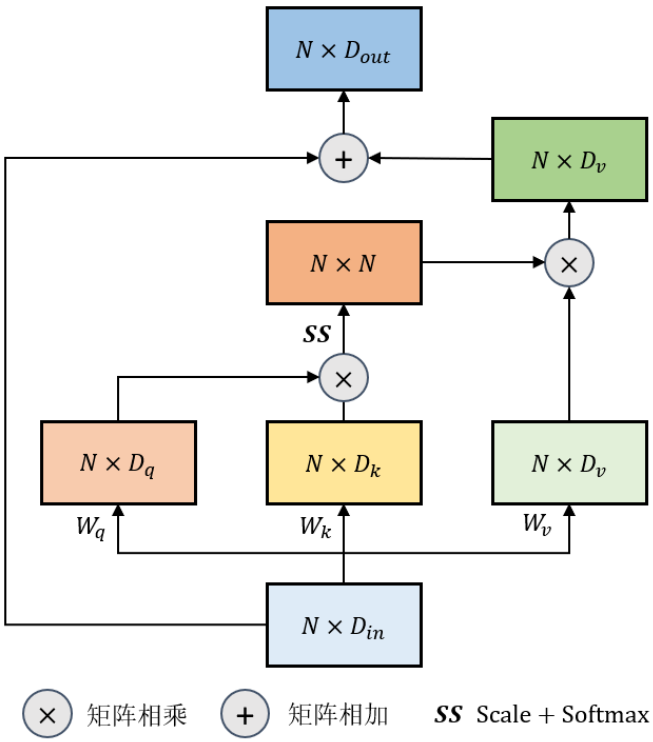

图 4-1 点云自注意力层

点云自注意力层的结构如图 4-1 所示，给定点云特征 $F_{in} \in R^{n\times d_{in}}$，根据自注意力机制，通过线性层或卷积层对输入点云进行变换，得到$Q \in R^{n\times d_q}$，$K \in R^{n\times d_k}$，$V \in R^{n\times d_v}$，见式(4-1)、(4-2)、(4-3)。其中$W_q \in R^{d\times d_q}$，$W_k \in R^{d\times d_k}$，$W_v \in R^{d\times d_v}$。

$$Q = F \cdot W_q \tag{4-1}$$

$$K = F \cdot W_k \tag{4-2}$$

$$V = F \cdot W_v \tag{4-3}$$

点云特征内关联性通过式(4-4)计算得出，Q与K进行矩阵乘法，计算点云特征之间的相似度，对该相似度进行缩放，应用 Softmax 进行归一化，得到注意力矩阵，由于Q与K进行矩阵乘法，因此d_q与d_k相等。V代表学习到的特征，注意力矩阵与V进行矩阵相乘，在矩阵层面对特征V加权计算，将特征关联性嵌入到特征中，得到注意力特征F_{sa}，如式(4-5)所示。对计算得到的注意力特征，为避免梯度消失和梯度爆炸问题，在注意力层后增加残差结构，如式(4-6)所示。

$$A = Softmax(\frac{QK^T}{\sqrt{d_k}}) \tag{4-4}$$

$$F_{sa} = \mathrm{A} \cdot V \tag{4-5}$$

$$F_{out} = F_{sa} + F_{in} \tag{4-6}$$

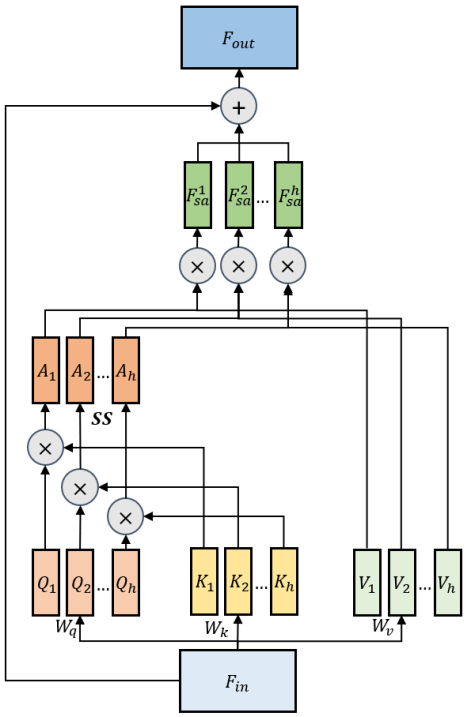

图 4-2 点云多头注意力

自注意力机制可以较好地提取出点云特征内关联性，然而注意力特征的单次计算不一定能够学习到所期望的特征。基于上述问题，在本节网络中引入多头注意力机制（Multi-Head Attention），对于点云特征内关联性，不同的注意力头可以计算出不同的关联特征。多头注意力机制采用多个注意力头，对于输入的特征$F_{in} \in R^{n \times d_{in}}$，生成多组$QKV$来分别计算注意力特征。

多头注意力机制如图4-2所示，假设当前注意力头数为h，输入点云特征为F_{in}，点数为n，特征维度为d_{in}，多头注意力首先计算整体的QKV，不再是计算一组QKV，如式(4-7)、(4-8)、(4-9)所示。此时，$W_q \in R^{h \times d \times d_q}$，$W_k \in R^{h \times d \times d_k}$，$W_v \in R^{h \times d \times d_v}$。

$$[Q_1, Q_2, \dots, Q_h] = F_{in} \cdot W_q \tag{4-7}$$

$$[K_1, K_2, \dots, K_h] = F_{in} \cdot W_k \tag{4-8}$$

$$[V_1, V_2, \dots, V_h] = F_{in} \cdot W_v \tag{4-9}$$

按照多头注意力机制的分组，计算多个头的注意力特征，如式(4-10)、(4-11)所示。得到每个头的注意力特征后，将所有特征拼接，并和输入相加得到F_{out}，如式(4-12)所示。

$$A_i = Softmax(\frac{Q_i K_i^T}{\sqrt{d_k}}) \tag{4-10}$$

$$F_{sa}^i = A_i \cdot V_i \tag{4-11}$$

$$F_{out} = concat(F_{sa}^1, F_{sa}^2, \dots, F_{sa}^h) + F_{in} \tag{4-12}$$

Simple TransformerNet 整体结构如图 4-3，其中编码器由输入嵌入（Input Embedding）、多头注意力（Multi-Head Attention）、LBR 层组成。

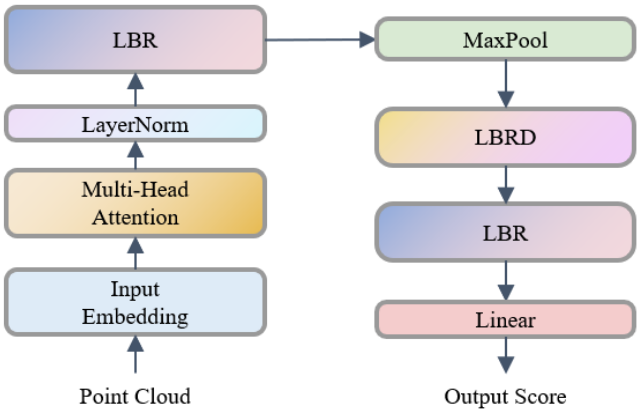

图 4-3 SimpleTransformerNet 结构

解码器采用 DGCNN[38]解码器结构，由三个线性层组成。点云作为三维空间数据，本身的坐标便具有丰富的位置信息，因此本节网络没有对点云进行位置编码，只采用输入嵌入将原始点云输入转换为编码特征。多头注意力模块由四个点云自注意力层组成，并在多头注意力后对输出特征进行 LayerNorm 正则化。图中 LBR 代表由线性层 Linear、归一化层 BatchNorm、激活函数 ReLU 组成的模块，LBRD 代表末尾增加 Dropout 层的 LBR 模块，输入嵌入模块由单层 LBR 实现。

4.2.2 基于特征距离的特征位置编码

当前点云学习领域内已有许多适用于点云的特征提取方法，一部分方法只对点云坐标进行处理，另一部分方法可学习点云的法线、颜色、等其他属性的特征，上述方法尽可能地提取点云的内在特征。此外，自 PointNet++提出后，采样与分组成为了点云特征提取和减少网络计算量的有效手段。点云采样得到关键点，经过分组后，点云被分为多个局部区域，对局部区域特征的提取是过去方法没有涉及到的。然而局部区域的特征仍然不能很好地表征点云内部的关联特征，点与点的联系被抽象到全局特征中，没有独立地被提取出来。受 DGCNN[38]的启发，将边卷积（EdgeConv）引入 TransformerNet 中。PCT 网络通过类 PointNet++的采样分组方式对特征进行学习，以点云空间位置为参考编码特征之间的关联。相较于 PCT，边卷积可以在高维特征空间内抽象点云的关联性，点与点之间的联系可以被直接抽象为图中的边，可以更好地抽象这种点与点之间的联系。因此本节基于特征距离的特征位置编码（Feature Distance Encoding）采用边卷积对点云特征进行抽象学习。

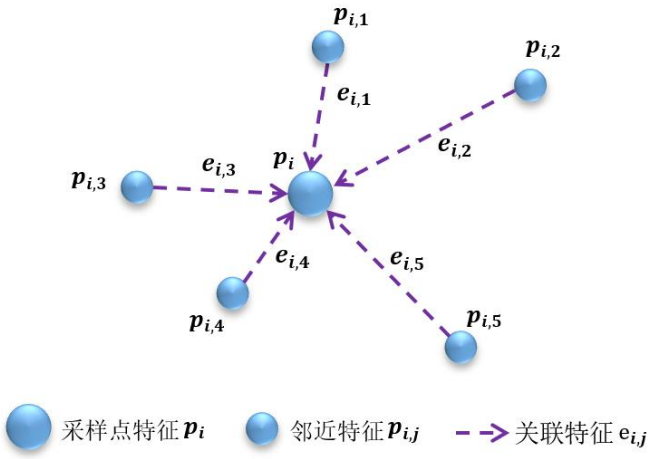

图 4-4 EdgeConv 图卷积

如图 4-4 所示，较大的蓝色点为关键点特征，较小的蓝色点为邻近点特征，点与点连接的紫色虚线代表关联特征。边卷积通过 KNN 算法对关键点和邻近点的特征进行查询，并将该局部区域构建为图，点即为图的节点，点与点的关联为图的边，最终对图的边进行卷积。假定输入点云为$P = \{p_1, p_2, \ldots, p_n\} \in R^d$，$d$为当前点云的特征维度，局部区域被表示为一个有向图$G = (\boldsymbol{V}, E)$，每个点的特征即为顶点，有向图的边$E$则由点与点的关联组成。更确切地说，对$P$中每个点$p_i$查询特征距离最近的$k$个点$p_{i,j}$，定义有向边为$E = \{e_{i,j}\}$，边特征$e_{i,j}$定义如式(4-13)所示。其中$h_\theta$采用一层感知机对局部区域特征进行提取，最终通过最大池化函数将局部特征聚合到当前点的特征上，如式(4-14)和(4-15)所示。

$$e_{i,j} = h_\theta(p_i, p_j - p_i) \tag{4-13}$$

$$e'_{i,j} = ReLU\left(\theta_m \cdot (p_j - p_i) + \phi_m \cdot p_i\right) \tag{4-14}$$

$$p'_i = \max_{(i,j)\in\Omega} e'_{i,j} \tag{4-15}$$

4.2.3 基于 Transformer 的点云学习网络

TransformerNet 网络结构如图 4-5 所示，网络分为分类网络和分割网络，图中左侧为编码器，中间是分割网络解码器，右侧是分类网络编码器。

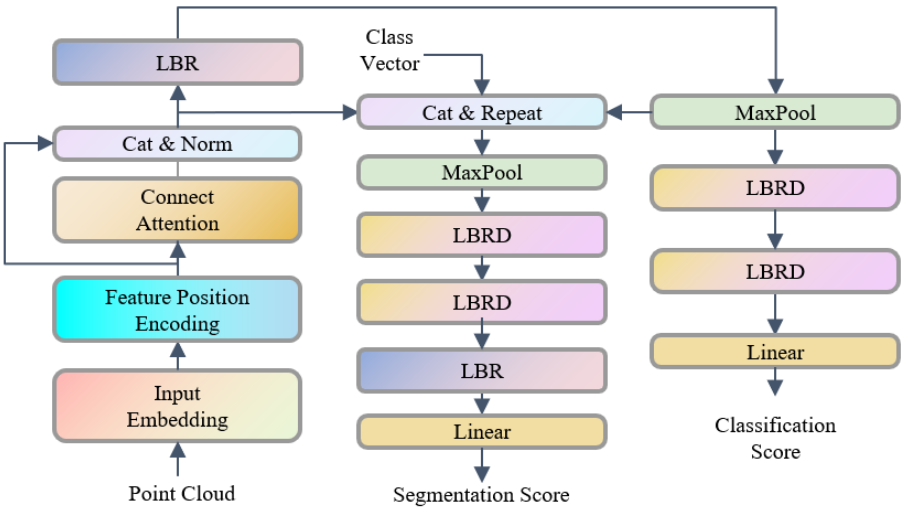

图 4-5 TransformerNet 网络结构图

TransformerNet 编码器由输入嵌入（Input Embedding）、特征位置编码（Feature Position Encoding）、联结注意力（Connect Attention）、拼接及正则化（Concat&Norm）和 LBR 组成。联结注意力由四层注意力（Attention, AT）层组成，受 PCT[62]的启发，将 L1 Norm 应用在 AT 层，因此 AT 层对特征先进行缩放，然后通过 Softmax 归一化特征的第一维度，最后通过 L1 Norm 归一化特征的第二维度，计算过程如式(4-16)、(4-17)、(4-18)、(4-19)所示。

$$\tilde{A} = \tilde{\alpha}_{i,j} = Q \cdot K^T \tag{4-16}$$

$$\bar{\alpha}_{i,j} = \frac{\tilde{\alpha}_{i,j}}{\sqrt{d^k}} \tag{4-17}$$

$$\hat{\alpha}_{i,j} = Softmax\left(\bar{\alpha}_{i,j}\right) = \frac{\exp\left(\bar{\alpha}_{i,j}\right)}{\sum_k \exp\left(\bar{\alpha}_{k,j}\right)} \tag{4-18}$$

$$\alpha_{i,j} = \frac{\hat{\alpha}_{i,j}}{\sum_k \bar{\alpha}_{i,k}} \tag{4-19}$$

四个 AT 层串联组成联结注意力模块，后紧跟 Concat&Norm 模块，将特征输入与输出相拼接和归一化，如图 4-6 所示。

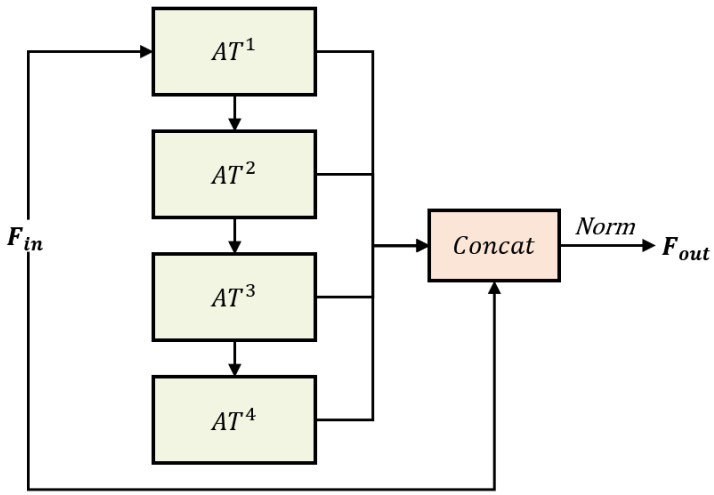

图 4-6 联结注意力与拼接模块

对于给定点云特征输入$F_{in} \in R^{n \times d_{in}}$，联结注意力中每层自注意力层的输出是下一层的输入，联结注意力计算过程如式(4-20)、(4-21)、(4-22)所示。

$$F_1 = AT^1(F_{in}) \tag{4-20}$$

$$F_i = AT^2(F_{i-1}), i = 2,3,4 \tag{4-21}$$

$$F_{out} = Norm\big(concat(F_{in}, F_1, F_2, F_3, F_4)\big) \tag{4-22}$$

归一化是避免网络过拟合的手段之一。对于输入的点云数据$F = f_{i,j,k} \in R^{b\times n\times d}$，其中$b$是 batchsize，$n$是点数，$d$为特征维度，LayerNorm 对每个点的特征进行归一化，计算每点特征的均值与方差，并对该点的特征进行归一化，如式(4-23)、(4-24)、(4-25)所示。在网络中自注意力机制对每个点的特征的内部关联性进行计算和提取，而 LayerNorm 对这种特征内部进行归一化，两者起到了相辅相成的作用。特征位置编码计算点与点之间的联系，而 BatchNorm 对一个批次内的所有点的特征进行归一化，因此在特征位置编码中应用 BatchNorm，避免关联特征的特征值出现极端值或梯度消失等问题。

$$\mu_{i,j}=\frac{1}{d}\sum_{k=1}^{\mathrm{d}} f_{i,j,k} \tag{4-23}$$

$$\sigma_{i,j}^2=\frac{1}{d}\sum_{k=1}^{\mathrm{d}} (f_{i,j,k}-\mu_{i,j})^2 \tag{4-24}$$

$$\hat{f}_{i,j,k} = \frac{f_{i,j,k}-\mu_{i,j}}{\sqrt{\sigma_{i,j}^2+\varepsilon}} \tag{4-25}$$

4.3 算法结果与分析

本章节根据前文介绍的模块和架构来建立三维点云学习网络，实现本章所介绍的 TransformerNet 网络，并采用 SimpleTransformerNet 作为 baseline 进行对照。将网络在物体分类、部分分割、场景语义分割三个任务上进行评估，分别采用 ModelNet40[22]数据集、ShapeNet[73]数据集、S3DIS[74]数据集进行实际的训练与测试。所用设备为两台机器，分类实验在机器 1 上完成，分割实验在机器 2 上完成，实验基础环境为：Ubuntu 18.04，Pytorch 1.7.0，详细环境如表 4-1 所示。

表 4-1 实验环境

机器编号	系统平台	CPU	GPU	显存	RAM	Cuda
1	Ubuntu 18.04	i7-8700	RTX 2080Ti	11GB	16GB	10.1
2	Ubuntu 18.04	i7-10700	RTX 3090	24GB	64GB	11.0

本研究采用三个在 3D 点云数据集中广泛使用的评价指标：总体精确度（Overall Accuracy，OA）、平均类别精度（Average Per-Class Accuracy，Avcc）、平均实例交并比（Instance mean IoU, Ins.IoU）和平均类别交并比（Class mean IoU, Cls.

IoU）。此外 mIoU 与 Ins.IoU 与相同，并在上一章节有所介绍，精度 Accuracy 的公式如式(4-26)所示，计算的是预测正确的样本数量占总样本数量的比例。

在点云分类和分割任务中，总体精确度 OA 和精度 Accuracy 的计算公式相同，分类任务中 TP 和 TN 代表预测正确的物体数量，分割时代表计算预测正确的点数量。平均类别精度 Avcc 公式如式(4-27)所示，其中 C 代表类别数量，$Accuracy_i$代表第i类的精度。

$$Accuracy=\frac{TP+TN}{TP+TN+FP+FN} \tag{4-26}$$

$$Avcc=\frac{1}{C}\sum_{i=1}^{C} Accuracy_i \tag{4-27}$$

在分割任务中，平均类别交并比 Cls.IoU 代表每个类别的平均交并比的均值，如式(4-28)所示，其中C代表类别数量，$mIoU_i$代表第 i 类的平均交并比。

$$IoU_{class}=\frac{1}{C}\sum_{i=1}^{C} mIoU_i \tag{4-28}$$

4.3.1 参数选择

为了提高 TransformerNet 网络的综合性能，本节在分类任务上进行广泛的实验，通过对比分析实验结果来挑选最优的参数值。本章网络最终的实验参数如表 4-2 所示，包括形状分类、部分分割、场景语义分割三类任务的基础参数。

表 4-2 实验基础参数

Task	Epoch	BatchSize	注意力头数	学习率	采样点	邻近点	优化器
形状分类	250	32	4	0.001	1024	20	SGD
部分分割	200	32	4	0.001	2048	40	SGD
场景语义分割	100	32	4	0.001	2048	20	SGD

（1）输入点云数量和邻近点数量对网络的影响

表 4-3 采样点与邻近点数量对比实验

Points	K	OA	Avcc
256	10	91.5	87.3
256	20	91.2	87.1
256	30	91.1	87.8
512	10	91.6	88.6
512	20	91.8	88.3

Points	K	OA	Avcc
512	30	91.8	88.5
1024	10	92.7	89.5
1024	20	**93.3**	**90.4**
1024	30	92.5	89.5

本小节探讨了采样点数量和邻近点数量对于模型的影响，在保证初始网络结构和其他参数不变的前提下，将采样点数量设置为 256、512、1024，邻近点设置为 10、20、30 进行对照实验。实验结果如图 4-3 所示，当采样点为 1024，邻近点为 20，网络的精度最高，OA 和 Avcc 分别达到了 93.3%和 90.4%，而随着采样点和邻近点的增加，精度并不会按照递增的趋势增加，而是一种先递增后递减的趋势。

为进一步分析实验结果，将表 4-3 的数据整理为气泡图，如图 4-7 所示，横轴代表采样点数量，纵轴代表邻近点数量，气泡大小与气泡值代表当前参数下的 OA。按照图 4-7 的趋势，当邻近点一定时，随着采样点数量增加，网络对于点云整体语义的预测更加准确，网络精度随之增加。当采样点为 1024，邻近点从 20 增加为 30 的情况下，网络的 OA 降低了 0.8%。纵向分析图 4-7，当采样点一定时，对于单个采样点来说，期望其邻近点是和该点语义相似的点，而当邻近点数量过多就会引入语义不相似的点，即邻近点数量过多会导致局部区域的语义污染，从而降低网络的预测能力。综上，采样点与邻近点的数量应该相匹配，本章选择采样点为 1024，邻近点为 20 作为后续实验的参数配置。

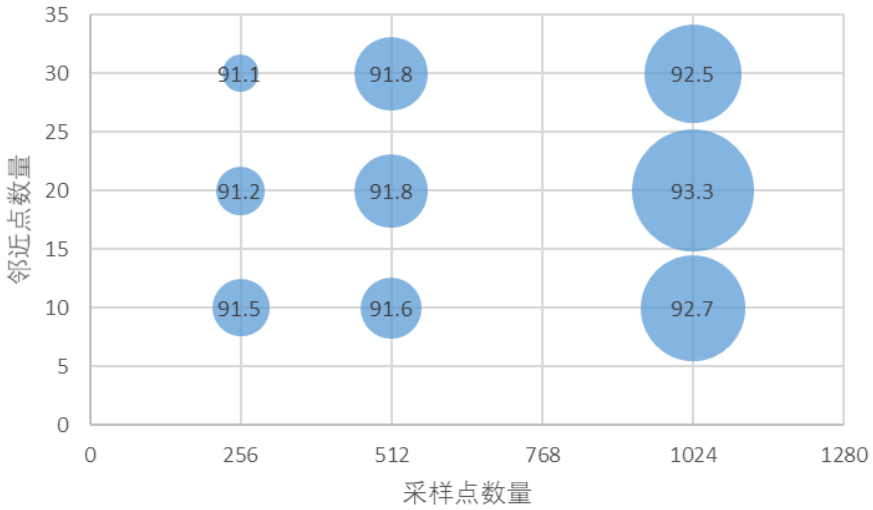

图 4-7 采样点与邻近点数量对比实验

（2）不同学习率对网络的影响

本小节探讨了学习率对于模型的影响，在保证初始网络结构和其他参数不变的前提下，设置学习率为 0.0001、0.0005、0.001 进行对照实验。实验结果如表 4-4 所示，表中数据均在同样的衰减策略和衰减系数下实验得来，分析可知学习率为 0.001 时网络呈现出最佳的表现，当学习率过低时，网络难以在相同的轮次内学习到更多的特征，当学习率过高时，网络的拟合难度增加，损失函数无法在短时间内快速降低。

表 4-4 学习率对比实验

Epoch	Learning Rate	OA	Avcc
250	0.0001	92.4	87.9
250	0.0005	93.0	90.2
250	0.001	**93.3**	**90.4**
250	0.005	92.0	87.2

4.3.2 形状分类

分类任务上选用 ModelNet40 数据集来评估本章节提出的网络，该数据集是 ModelNet 的子集，ModelNet 于 2015 年被普林斯顿大学发布，共有 662 种目标分类，127915 个 3D CAD 模型。ModelNet40 数据集包含 40 个类别共 12311 个 3D CAD 网格模型。本章采用与 PointNet 相同的设置，其中 9843 个模型用于训练，2468 个模型用于测试，图 4-8 可视化出 ModelNet40 部分模型。

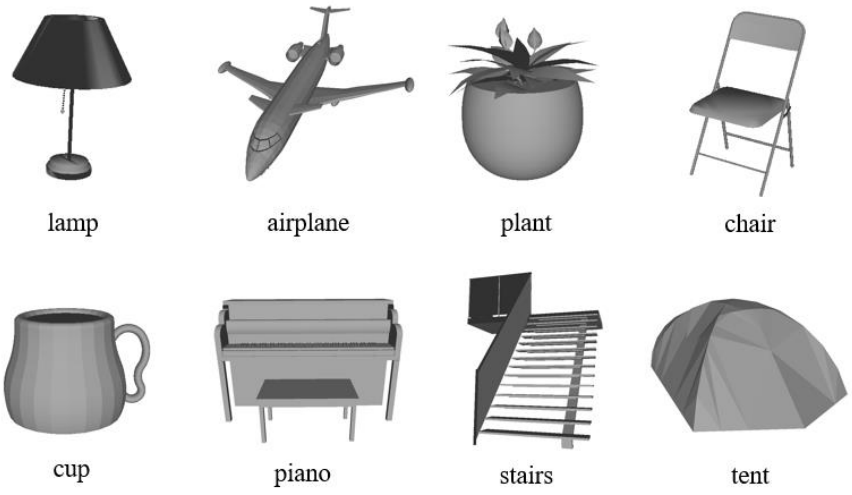

图 4-8 ModelNet40 部分模型

分类网络的编码部分采用通用的 TransformerNet 编码器，其中每层自注意力模块输出维度为 64，多头注意力模块将输入与四个自注意力模块的输出相拼接，得到 320 维度的输出，并通过一个多层感知机训练为 1024 维点云全局特征；解码器采用 DGCNN 相同的结构，编码器得到的特征通过最大池化和三个多层感知机得到预测结果，每层输出维度分别为 512、256、40，其中 40 是分类类别。

实验在每个网格模型上均匀采样 1024 个点并缩放为球体以供训练，此外对点云坐标以 0.6 至 1.5 倍进行缩放，在[-0.2, 0.2]范围内进行平移，并且打乱点云数据的顺序来扩充数据集，提高模型的鲁棒性。训练总轮次 Epoch 设置为 250，batchsize 设置为 32，在训练过程中采用随机梯度下降优化器（Stochastic Gradient Descent, SGD）来更新学习率，初始学习率设置为 0.001，衰减系数设置为 0.0001，对输入的点云，通过 k 邻近搜索 20 个点进行特征提取。

表 4-5 形状分类实验结果

Method	Input	Points	OA
PointNet[33]	P	1k	89.2
A-SCN[75]	P	1k	89.8
Kd-Net[25]	P	32k	91.8
PointNet++[34]	P	1k	90.7
PointNet++[34]	P, N	5k	91.9
PointGrid[28]	P	1k	92.0
PointWeb[44]	P	1k	92.3
PointCNN[43]	P	1k	92.5
PointConv[72]	P, N	1k	92.5
KPConv[39]	P	7k	92.9
DGCNN[38]	P	1k	92.9
RS-CNN[41]	P	1k	92.9
PCT[62]	P	1k	93.2
LSANet[53]	P	1k	93.2
PointTransformer[61]	P	1k	93.7
DTNet[76]	P	1k	92.9
Point-BERT[77]	P	1k	93.2
ACET[64]	P	1k	93.4
baseline	P	1k	90.9
TransformerNet	P	1k	93.3

ModelNet40 分类结果如表 4-5 所示，与当下主流网络在评估结果上进行了比较。以整体精度 OA 作为评价指标。从表中可以看到，TransformerNet 在分类任务

上取得了优异的成绩。相比于主流的 PointNet、PointNet++、PointCNN、DGCNN 等点云学习网络，本章网络也都有着一定的提升，TransformerNet 通过注意力机制更好地将局部特征和全局特征从点云数据中抽取出来，可以达到很高的精确度。和同样应用了注意力机制的网络相比较，TransformerNet 首先将 Transformer 结构应用到点云学习上，而 LSANet 更为直接，通过点云位置信息或特征信息训练出注意力分数矩阵，并和特征进行矩阵元素相乘。Transformer 结构可以让局部点云学习到该局部的关联信息和语义信息，这也是自注意力的优点，而相比于 PCT 网络，TransformerNet 的优点在于采用了特征位置编码，当点云特征无法较好地被抽象时，将点与点的关系转换为特征，并对该特征进行学习。PointTransformer 超过本章方法 0.4%，这是由于前者在点集上应用了层次化结构，并将全局特征和局部特征结合预测。从实验结果看，本章网络在分类评估中优于大部分主流方法，证明了 TransformerNet 分类网络结构的有效性和优异性。

4.3.3 部分分割

ShapeNet 数据集是斯坦福大学发布的大规模三维 CAD 模型数据集，模型类别如图 4-9 所示。本节选用该数据集的部分分割子集进行实验和评估，该子集有 16 种点云形状，共计 16881 个点云数据，其中每种形状可分为 2~5 个部分，共有 50 个部分类别。

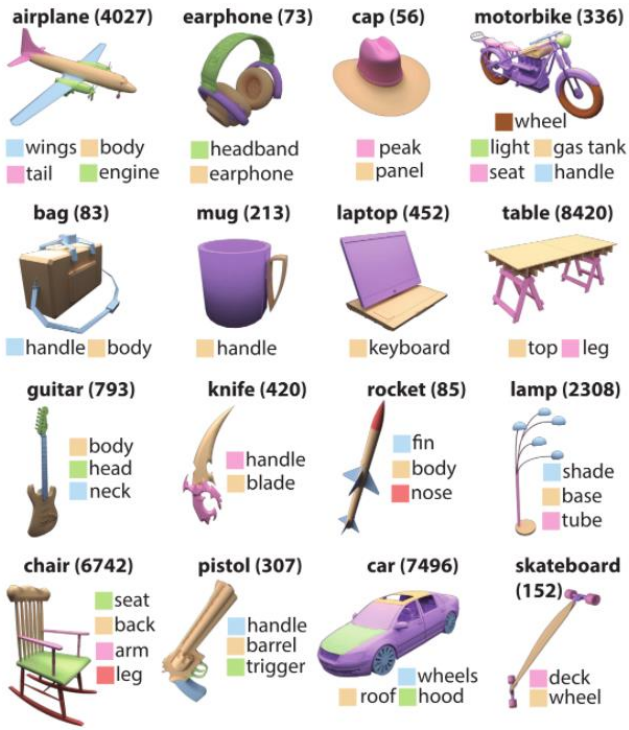

图 4-9 ShapeNet 模型图[73]

部分分割网络的编码部分采用 TransformerNet 编码器，包括输入嵌入、特征位置编码、联结注意力和一个 LBR 层，解码器采用 DGCNN 相同的结构，TransformerNet 编码器得到全局点云特征，其中 Transformer 结构的每个注意力头的输出为中间点云特征，将分割类别 One-Hot 编码向量与上述两者拼接起来，最后通过四层 MLP 得到分割结果，其输入维度分别为 1408、256、256、128。

数据集的处理采用与 DGCNN 相同的设置，在训练过程中将点云采样为 2048 个点输入训练，邻近点数量设置为 20，训练的总轮次设置为 200，其余实验参数与分类实验一致。

表 4-6 部分分割实验结果

Method	3DCNN[33]	Kd-Net[25]	PointNet[33]	PointNet++[34]	TransformerNet
Ins.IoU	79.4	82.3	83.7	**85.1**	84.2
Cls.IoU	-	-	80.4	**81.9**	80.6
airplane	75.1	80.1	**83.4**	82.4	82.5
bag	72.8	74.6	78.7	**79.0**	75.6
cap	73.3	74.3	82.5	**87.7**	83.1
car	70.0	70.3	74.9	**77.3**	76.0
chair	87.2	88.6	89.6	**90.8**	90.2
earphone	63.5	73.5	**73.0**	71.8	72.9
guitar	88.4	90.2	91.5	91.0	91.2
knife	79.6	87.2	85.9	85.9	**87.9**
lamp	74.4	81.0	80.8	**83.7**	83.5
laptop	93.9	94.9	95.3	95.3	**95.6**
motorbike	58.7	57.4	65.2	**71.6**	64.3
mug	91.8	86.7	93.0	**94.1**	94.0
pistol	76.4	78.1	81.2	81.3	**80.6**
rocket	51.2	51.8	57.9	58.7	**59.1**
skateboard	65.3	69.9	72.8	**76.4**	72.0
table	77.1	80.3	80.6	**82.6**	81.6

表 4-6 对网络精度进行分析，并对经典的点云语义分割网络进行了对比分析。PointNet 是基于点的网络的开山之作，3DCNN 是 PointNet 工作中的 baseline，具体做法是体素上进行 3D 的卷积并得到分割分数。TransformerNet 的 mIoU 达到了 84.2%，高于 3DCNN 和 PointNet，这是由于本章网络引入了 Transformer 结构，在特征提取能力上比上述两个网络更加强大，可以根据整体点云信息得到每个点在整体中的高维度信息。此外，本章网络在 knife、laptop、pistol、rocket 类别上取得了较为优异的表现。然而本章网络在整体的平均交并比上与 PointNet++仍相差

0.9%，这是因为 PointNet++增加了采样和分组的机制，在多尺度上对点云的局部特征进行了提取，而本章网络虽然也有邻近点的选取，但是 Transformer 结构随着数据过大，提取样本之间的关联关系的能力也会变差，例如表 4-3 中，邻近点从 20 增加到 30 反而会降低模型的能力。

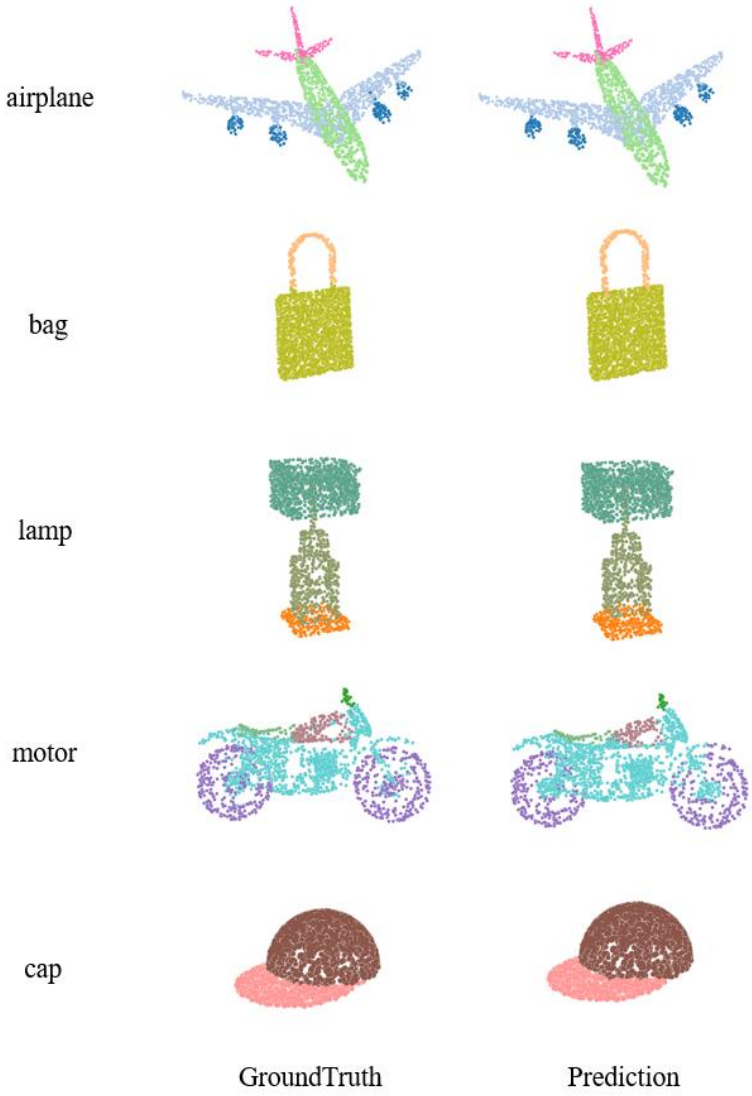

图 4-10 ShapeNet 分割结果可视化

图 4-10 为 ShapeNet 部分模型的可视化分割结果，图中每一行表示一个物体的分割结果，第一列代表数据集中的真实标签可视化，第二列为网络预测结果可视化。由图 4-10 可以看出，本章提出的网络在 ShapeNet 数据集上的表现优秀，与真实标签非常接近。然而，该网络仍旧有一定的缺陷，正如前文所说，邻近点的特征在一

定程度上会影响关键点的特征，例如图中摩托的分类结果，在摩托的轮胎附近，由于车架和轮胎十分接近，导致部分真实标签为轮胎的点被网络错误分类为车架。

4.3.4 场景语义分割

S3DIS 是斯坦福大学发布的大规模室内空间数据集，包含三个不同建筑的六个区域，11 种场景，共有 271 个独立房间，场景模型如图 4-11 所示。数据集包括 2D 数据、3D 数据和 2.5D 数据，在本章节中只对 3D 点云数据进行处理和评估。S3DIS 三维点云数据具有实例级语义和几何注释，在场景语义分割任务中共有 13 个标签，包括桌子、沙发、椅子、窗户、门等，场景中每个点都被标注为特定的标签。该数据集的数据包括点云的位置信息、颜色信息、法线向量等，此外还包括场景的相机信息和每个扫描的位置信息。

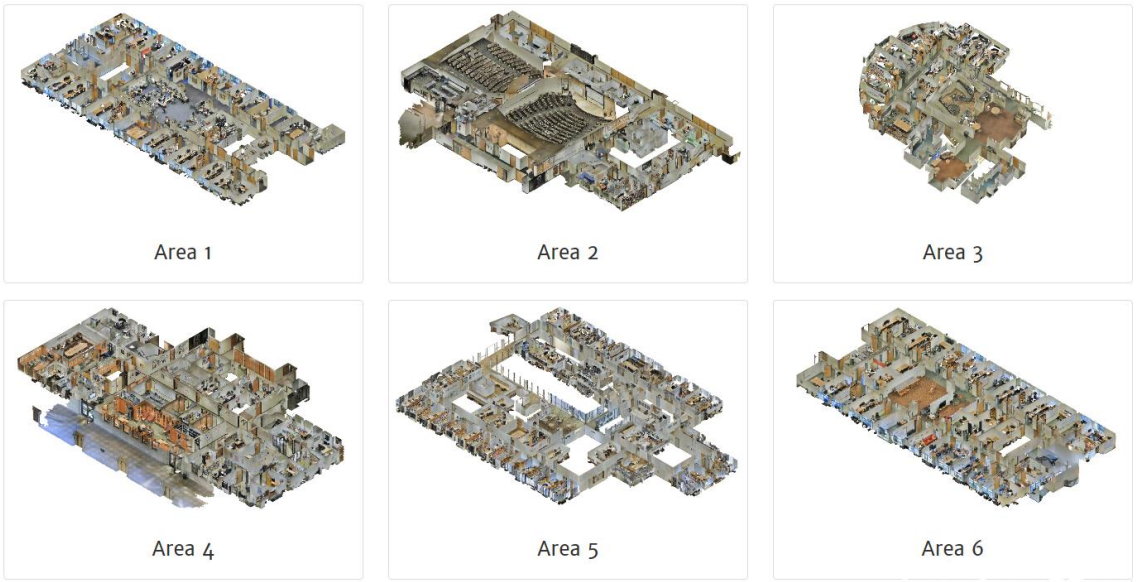

图 4-11 S3DIS 区域模型图

语义分割网络架构如图 4-5 所示，网络的编码部分采用通用的 TransformerNet 编码器，解码器采用 DGCNN 相同的结构，与部分分割的网络结构类似，编码器得到的特征首先经过最大池化得到一个全局的特征向量，接着将全局特征向量重复后和编码器训练过程中的每个点的特征向量相拼接，最后经过三层线性层进行解码，三层线性层的输出维度分别为 512、256、13，并且包括 BatchNorm 和 ReLU，最终得到预测结果。

本节采用与 PointNet 相同的设置，将场景分割成 1m×1m×1m 的立方块输入网络进行训练，输入点云的维度为 9，包括点云位置信息、RGB 信息和在训练时输入点云空间坐标 、颜色和归一化的空间坐标。实验过程中，不同于 PointNet 和

DGCNN，由于显存限制，在每个立方块上采样 2048 个点进行训练，测试时对所有点进行语义预测评估。对于数据集中六个区域，采用 6-fold 交叉验证方法对所有区域进行综合评估。每次实验采用 5 个区域作为训练集，余下 1 个区域作为测试集，将每一个区域都作为测试集进行训练，得到六次实验结果，充分利用数据集的数据，提高网络模型的泛化能力。在训练过程中采用随机梯度下降优化器（SGD）来更新学习率，每个场景训练的总轮次 Epoch 设置为 100，对输入的点云，通过 k 邻近搜索 20 个点进行特征提取，其余实验设置与分类实验一致。

表 4-7 场景语义分割实验结果

Method	Input	OA	mIoU
PointNet[33]	4096 × 9	78.6	47.7
PointNet++[34]	4096 × 9	81.0	54.5
PointCNN[43]	4096 × 9	88.1	65.4
3D-RNN[78]	6400 × 6	86.9	56.3
DGCNN[38]	4096 × 9	84.1	56.1
RSNet[41]	4096 × 9	-	56.5
RandLA-Net[36]	4096 × 9	88.0	70.0
baseline	2048 × 9	83.9	57.2
TransformerNet	2048 × 9	85.9	60.6

总体精度和 mIoU 评估结果如表 4-7 所示，该表将本章网络与近年来的经典点云语义分割网络进行了对比分析，TransformerNet 的 OA 和 mIoU 分别达到了 85.9% 和 60.6%，高于大部分网络，其中 mIoU 相比于 PointNet 高出了 12.9%，对于 PointCNN 和 RandLA-Net 相差分别 5.4%和 9.4%，综合对比，TransformerNet 在空间特征上可以提取到较为准确的信息，对于局部的空间结构拥有较好的预测能力，但是由于显存限制，采样点数量无法以 4096 输入网络，4096 点的输入点云对于显存要求巨大。然而，将采样点数量减小为一半后，输入点云的信息会所丢失，造成精度下降。此外，点云场景分块输入也是造成错误预测的因素，如 RandLA-Net[36] 所述，当点云场景分块后，单个类别的物体会被分为多个块，在每个块中预测该类别会受到其他类别并且相邻的点的影响。

除定量分析外，图 4-12 展示了可视化结果用于定性分析，可以看出网络对于几何结构明显，且体积较小的物体的分割能力是较强的，例如场景中红色的椅子、粉红色的桌子等，但是某些结构性物体，例如天花板、门、墙壁、窗户等，这些物体往往与其他物体有大面积或者体积的相邻，并且本身具有二维特性，没有突出的三维结构，网络在对场景进行分块时，容易将此类物品进行错误预测，这也说明了

本章提出的网络在二维形状的物体上泛化性较差。本章网络主要的优点在于可以捕捉局部几何信息，有效区分几何特征明显的物体。

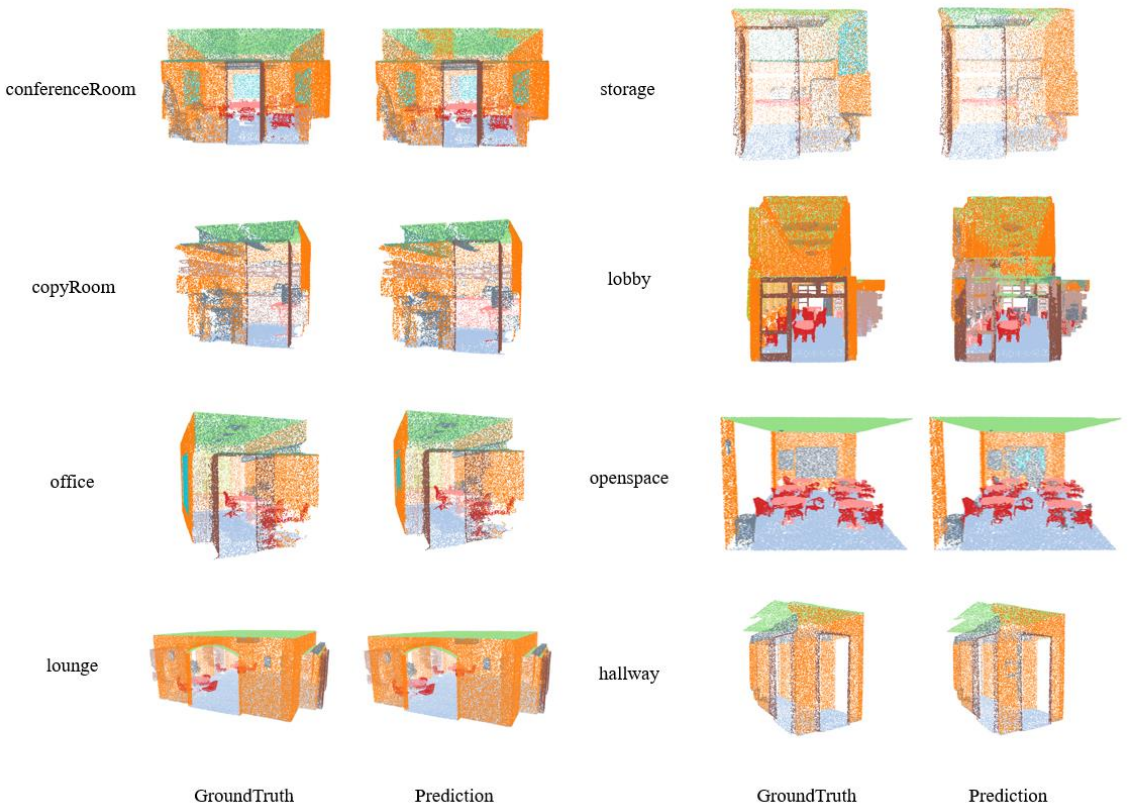

图 4-12 S3DIS 分割结果可视化

4.3.5 综合实验

为进一步分析本章网络的效果，评估不同情况下的网络表现，本节在 ModelNet40 数据集上通过多项对照实验对网络模块设计和网络健壮性进行验证。

（1）不同模块对网络的影响

本章网络主要模块和技术包括特征位置编码、联结注意力、L1-Norm。为了验证上述模块的有效性，对集成不同模块的 TransformerNet 进行消融实验。表 4-8 评估了本章网络模块的有效性，baseline 为 Simple TransformerNet，分别进行了单头注意力和多头注意力的实验，所比较网络的解码器均为多层 MLP 组成。由表 4-8 可知，单头注意力结构的表现比 PointNet 的效果更好，说明了注意力结构强大的特征提取能力。在 TransformerNet 的不同模块实验中，未使用的模块以 baseline 中对应模块代替，例如只包括特征位置编码模块的 TransformerNet 采用和 baseline 相同的四头自注意力模块。

表 4-8 网络模块消融实验

方法	特征位置编码	联结注意力	L1-Norm	OA	Avcc
PointNet				89.2	86.2
baseline(one head)				90.4	85.4
baseline				90.9	86.6
TransformerNet	√			92.7	89.0
TransformerNet	√	√		93.0	90.1
TransformerNet	√	√	√	93.3	90.4

由表 4-8 可知，本章提出的网络在集成全部模块时表现出最高的精度，与 PointNet 相比，本章 baseline 高出 1.7%，本章提出的网络高出 4.1%。此外，与 baseline 相比较，集成不同模块的 TransformerNet 也有着相应的提升，其中特征位置编码提升最大，精度为 1.8%。其他对比可进一步证明其他模块的有效性，其中联结注意力相比基础的自注意力机制有更强的特征学习能力，L1-Norm 可以有效地提高网络表现，防止网络出现过拟合等现象。

（2）不同特征编码信息对网络的影响

特征位置编码将点云的关联性充分编码到特征中，在该模块中，选择合适的编码信息是至为重要的，在 4.2.2 节中介绍到关联特征可通过关键点和邻近点的特征表示，对于上述特征的选择，在表 4-9 中给出了对比结果。表 4-9 中，Fi 代表关键点特征，Fj 代表邻近点特征，Fj – Fi 代表归一化的邻近点特征。表中给出了四种编码方式，可以看出将归一化的邻近点特征和关键点特征相拼接的表现是最好的，与编码关键点特征的方式高出 1%。其中编码 Fj – Fi 的表现要优于编码 Fj 的表现，这是由于邻近点的特征有可能在数值上相差巨大，当邻近点的特征向量通过关键点进行归一化后，特征值的分布趋于稳定，避免了差距较大的邻近点特征对网络学习造成影响。

表 4-9 特征编码信息对比实验

特征编码信息	OA	Avcc
Fi	92.3	88.4
Fj	92.3	88.6
Fj - Fi	92.5	89.3
(Fj – Fi, Fi)	93.3	90.4

（3）不同池化方式对网络的影响

网络解码器作为网络的预测模块，对网络的效果起到重要的作用，表 4-10 给出了本章网络的解码器在不同池化方式下的实验结果。池化是对点云特征提取的

有效方式，同时池化作为一种对称函数，可以有效解决点云的无序性问题。在表 4-10 中，对比了最大池化和平均池化和其组合的四种方式，其中最大池化表现最优，平均池化表现最差。在分类任务上，最大池化可以保留当前点云数据最显著的特征，与点云分类任务的本质相同，因此有着最好的表现，而平均池化弱化了点云特征中突出的特征值，网络的预测能力因此下降。同样的，在两种池化方式相加和相拼接的实验中，拼接方式保留了点云的明显特征，但也引入了平均池化特征，从而对网络预测带来了干扰，相加的方式同时弱化了最大池化，强化了平均池化，使得该方式的表现优于平均池化，而劣于最大池化。

表 4-10 池化方式对比实验

池化方式	OA	Avcc
max	93.3	90.4
avg	91.6	88.2
max+avg	93.0	88.8
(max, avg)	93.1	90.3

（4）不同归一化方式对网络的影响

表 4-11 归一化方式对比实验

AttentionScore	FeatureScore	OA	Avcc
-	LayerNorm	93.3	90.4
-	BatchNorm	92.5	88.8
LayerNorm	LayerNorm	92.7	89.1
BatchNorm	BatchNorm	92.5	88.7
LayerNorm	BatchNorm	92.7	89.7
BatchNorm	LayerNorm	92.7	89.5

在 4.2.3 节中，对自注意力机制进行了优化，采用了 LayerNorm 作为输出特征归一化方法，在表 4-11 中给出了自注意力模块中不同归一化方法的实验结果。其中 AttentionScore 代表对注意力分数矩阵归一化，FeatureScore 代表对自注意力模块的输出特征归一化，归一化维度均为特征维度。由表 4-11 可知，无论哪种组合方式，对注意力分数矩阵进行归一化会减弱该矩阵语义修正的效果，造成网络的精度下降。由于自注意力机制学习的是点云特征内部的关联性，LayerNorm 归一化的维度在特征维度，仅仅以单个点云的特征归一化，也就使得特征内部的关联性分布更加稳定，易于网络学习。而 BatchNorm 方法归一化每个批次的特征，将特征内

部的关联性打乱，将每个批次的数据接近该批次数据的分布进行，也就弱化了网络已学习到的特征，抑制了网络学习能力。

4.4 本章小结

本章内容由三部分组成。第一部分介绍了点云特征提取的背景，包括点云数据的特点，难以提取点云的关联特征的原因，以及在自然语言处理领域的 Transformer 方法。针对点云数据与序列数据的相同点和不同点提出了点云适应化的编码器。

第二部分介绍了本章算法的整体结构和具体细节，首先提出自注意力机制在点云分类网络上的基础实现。然后介绍了特征位置编码的思想，通过充分捕捉点云的关联性来增强特征学习能力。最后详细说明了本章提出的基于 Transformer 的点云学习网络结构，并指出了本章网络的在前文点云分类网络基础上的优化。

第三部分设计多组实验来验证本章算法。首先讲解了本章网络的实验环境和实验设置，并在实验超参数上进行了多项实验，筛选最优参数，为后续的实验打好了基础。其次分别在点云形状分类、部分分割、场景语义分割上进行实验和分析，在不同数据集上的表现证明了基于 Transformer 的点云分割网络的优越性。通过将分割结果可视化，展示了本章算法在数据集上的分割效果。最后对本章网络的各个模块设计了消融实验和对比实验，在不同方法的对比下论证了本章模块的有效性。

第五章 基于通道自注意力的点云分割网络

5.1 引言

点云的优点在于其包含丰富的三维结构和空间信息，并且点云作为一种无序的数据，点云的输入十分简单，即点的向量组成的矩阵。丰富的三维结构也为点云学习带来了挑战，点云具有无序性和旋转不变性，更为重要的是点云是稀疏的，由于卷积需要规则化的数据，无法直接应用在点云数据上。为了解决上述问题，研究人员提出了许多方法[79]，包括基于多视图的方法、基于体素的方法、基于点的方法等。其中最直接的方法为 PointNet 网络，其采用多层感知机学习点云的隐含信息，并采用对称函数得到可描述点云的全局特征。点云的空间信息包含了点与点的关联信息，PointNet++将点云的特征信息描述为局部特征和全局特征，网络学习到的特征本质上仍然是全局特征，并没有提取出点与点的关联信息。

随着点云采集设备的成熟化和普及化，点云的数据量日益增长，现有数据集中包含海量的点数据，网络可使用的数据使得网络变得臃肿庞大。同时，点云学习方法也日益复杂，其中三维卷积对计算资源的需求是巨大的。对于神经网络而言，数据量越大、数据分布越规范，网络可以学习到的特征也就更加准确。对前文提出的基于 Transformer 的点云学习网络进行分析，该网络可以较好地提取点与点的关系和点的特征间关系，但是计算量较大，训练时间缓慢，并且对全局关联关系进行学习时没有考虑到局部特征的学习。

本章基于以上问题，本章在 Transformer 网络和 PointNet++提出的层次化结构的基础上，经过不断重构和优化，重新构建了一种基于通道自注意力的点云分割网络（ChannelAttentionNet, CANet），总结来说，本章的主要贡献如下：

(1) 设计了点云局部特征抽象模块（Local Feature Abstraction, LFA），对点云的坐标及特征进行聚合，通过层次化结构对点云进行采样和分组，改善局部特征提取，且可以降低网络输入，进一步减小网络参数量。

(2) 改进自注意力机制，优化自注意力机制的计算方法，降低网络计算量，并巧妙的将优化改进后的通道自注意力模块（Channel Self-Attention, CSA）应用到到点云分割网络中的每个点云局部特征提取模块之后，增强网络的学习能力。

(3) 本章对邻近算法进行了讨论，采用基于余弦距离的 K 邻近算法，不同于常规的 KNN[80]算法按照欧氏距离选取邻近点，本文点云特征相似度算法通过余弦距离选取相似特征，将特征空间内的相似特征进行聚合，辅助特征提取模块进行学习。

5.2 算法工作

本章基于通道自注意力的点云分割网络（CANet）采用自注意力机制，并通过采样和分组的方法对点云局部特征进行学习，网络算法流程如图 5-1 所示。CANet 由编码器和解码器组成，编码器包括局部特征抽象模块、通道自注意力模块组成，解码器多个全连接层（Full-Connected Layer，FC）组成。

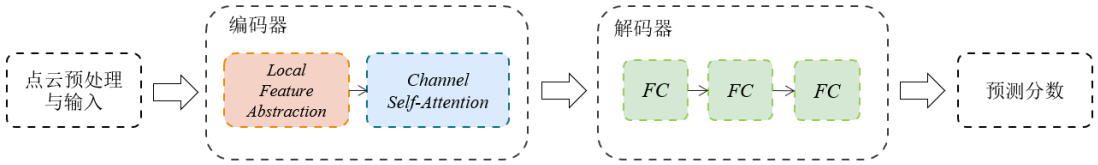

图 5-1 CANet 网络流程

如图 5-1 所示，网络工作流程可表述为点云自编码器开始进行特征提取，然后解码器将特征解析为预测分数的过程。编码器中，输入点云通过多个由局部特征抽象模块和通道自注意力组成的模块。在局部特征抽象模块中，通过最远点采样（Farthest Point Sampling，FPS）选出点云的关键点，这些关键点可以表征整体点云，点云的采样减少了网络需要处理的数据大小，随后采用点云特征相似度算法对关键点的邻近点进行分组，得到一系列关键点和邻近点，形成点云的局部区域，局部区域的特征由共享多层感知机（Shared-MLP）抽象得到。经过局部特征抽象后，局部区域的特征经过通道自注意力模块，该模块对局部特征进行进一步学习，对特征间的关联关系进行提取。

算法 5-1 TransformerNet++网络算法伪代码

算法：基于通道自注意力的点云分割网络
输入：点云数据 B x 3 x C
参数：迭代次数 epoch，学习率 lr，采样点数N_i，邻近点数K_i
输出：预测分数 B x N x num_class
算法流程：
1. for t←1,2…T, do
2. LFA 模块对点云进行采样分组，提取局部区域特征，得到$B \times N_i \times D_1$的特征
3. CSA 模块对局部区域特征进行学习和语义修正，得到$B \times N_i \times D_2$的特征
4. 重复步骤 2 和步骤 3，得到增强后的特征
5. 网络解码器将特征解析为预测分数
6. 计算网络损失函数，反向传播更新网络参数
7. End for
8. 输出网络训练结果，将网络模型保存在本地

解码器对编码器得到特征进行维度的转换，将抽象后的特征通过神经网络转换为预测分数，其中分类任务转换为单个类别的预测分数，分割任务转换为每个点的预测分数，本章提出的网络计算过程如算法 5-1 所示。

5.2.1 点云局部特征抽象

如图 5-2 所示，本章局部特征抽象模块主要包括最远点采样和 K 紧邻分组操作。假设局部特征抽象模块的输入为$N \times D$的点云特征$\{F_i | i = 1,2, \dots, N\}$和$N \times 3$的点云位置信息$\{P_i | i = 1,2, \dots, N\}$，其中$N$表示输入点云的点个数，$D$表示输入点云的特征维度。对于初始点云数据，特征维度一般为 3，表示点云的三维坐标(x, y, z)，特定的点云数据会包含颜色、法线等信息，例如点云(x, y, z, r, g, b)代表包含 RGB 颜色的数据。

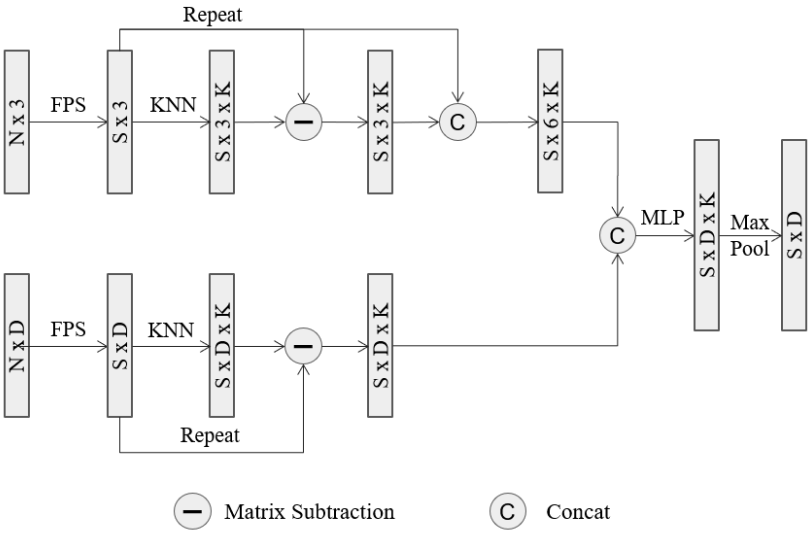

图 5-2 局部特征抽象模块

最远点采样可以将点云中两两距离最远的点通过迭代计算得出，求出的点集可以近似代表整个点云，在局部特征抽象模块中，通过最远点采样求得可以代表整体点云的关键点，将点云的分辨率和数据量降低至 S，即采样的关键点数量。将关键点记为p_i，在下一步中 K 近邻算法求得每个关键点的邻近点，例如p_i^j即为p_i的第j个邻近点。此外，最远点采样不是必选流程，局部特征抽象模块可以不使用最远点采样，直接进行下一步的 K 近邻分组操作，这时 K 近邻分组对整体点云进行查询，即对整体 N 个点中的每个点来查询其 K 个邻近点。

不同于 PointNet++网络，本章网络在对特征进行分组时，并不采用查询位置信息的邻近点来分组特征。受到 DGCNN 的启发，在 KNN 邻近算法中引进特征相似

度算法，局部特征抽象模块采用余弦相似度作为特征的比较算法，在特定的特征空间内查询特征之间的邻近程度，在实际计算中查询特征之间的余弦距离。将采样后关键点的特征记为f_i，通过特征相似度查询到的邻近特征记为f_i^j，通过特征相似度将特征空间中语义相似的点进行聚合，并将邻近特征与关键点特征相减进行归一化。最终将关键点的位置信息、归一化的邻近点位置信息和归一化的邻近特征拼接在一起，通过一层多层感知机对上述特征进一步学习，局部特征抽象模块的输出F_{out}如式(5-1)计算得出，其中 MLP 为多层感知机，Concat 代表拼接操作。

$$F_{LFA} = MLP\left(Concat\left(p_i, p_i^j - p_i, f_i^j - f_i\right)\right) \tag{5-1}$$

5.2.2 通道自注意力机制

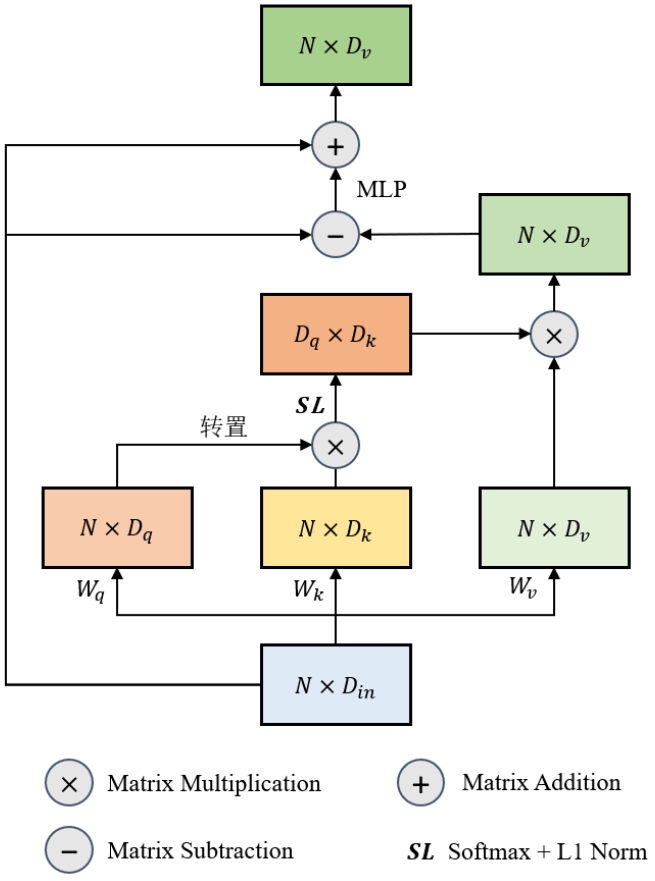

图 5-3 通道自注意力机制模块

由 4.3 节实验可知，Transformer 结构中的自注意力机制在点云学习上同样具有强大的特征提取能力，然而在点云这种海量的数据上，自注意力机制对硬件和计算资源的需求也十分庞大。在上一节中，局部特征抽象模块通过对点云下采样，降低点云数量，从而减小网络计算量，本节对自注意力机制进行分析，并提出一种通

道自注意力机制，在自注意力模块上降低模型的计算量。通道自注意力机制参考了PCT[62]网络，对其提出的偏移注意力（Offset-Attention）进行了改进。如图 5-3 所示，通道自注意力和自注意力的不同在于注意力分数矩阵的计算，自注意力机制根据输入特征得到Q、K、V，然后将Q与K的转置进行相乘得到注意力分数矩阵A，如式(5-2)所示。

$$A_{CSA} = Softmax(KQ^T) \tag{5-2}$$

不同于上述做法，通道自注意力机制计算注意力分数矩阵的过程如式(5-2)所示，将K与Q的转置进行矩阵相乘，在本节模块中$Q \in R^{n\times c}$，$K \in R^{n\times c}$，$V \in R^{n\times c}$，因此由式(5-2)计算得到的$A_{CSA} \in R^{c\times c}$，而式(4-4)计算得出的$A \in R^{n\times n}$，在后续操作中，注意力分数矩阵与$V$进行矩阵乘法，完成特征的注意力运算，通道自注意力模块的计算如式(5-3)所示，第四章中自注意力模块的计算如式(4-5)所示。对上述两种计算方式进行分析，可知自注意力模块的时间复杂度和空间复杂度分别为$O(n^2c)$和$O(n*c+n*c+n*n)$，而通道自注意力模块的时间复杂度和空间复杂度分别是$O(nc^2)$和$O(n*c+n*c+c*c)$。因此当两种模块中的$c \ll n$时，网络的计算时间和占用的空间大小会有一定的减小，优化了网络的效率。

$$F_{CSA} = V \cdot A_{CSA} \tag{5-3}$$

通道自注意力模块中，参考偏移注意力引入了位置编码，具体来说，模块输入特为$F_{in} \in R^{n\times c}$，将该点集的位置信息$P \in R^{n\times 3}$同时输入，并编码到特征中，具体做法如式(5-4)所示，模块输入被更正为位置编码与局部特征抽象模块输出的和，类似于残差结构，引入位置编码使得网络最差也可以学习到位置编码的特征，防止网络过拟合。

$$F_{in} = F_{LFA} + MLP(P) \tag{5-4}$$

在本节提出的 CSA 模块上，引入多头机制，将多个通道自注意力模块串联在一起，级联输入输出，并将每个模块的输出拼接在一起，得到最终的多头通道自注意力（Multi-Head Channel Self-Attention）的输出，计算公式如式(5-5)至(5-7)。

$$F_1 = CSA_1(F_{in}) \tag{5-5}$$

$$F_i = CSA_1(F_{i-1}), i = 2,3,4 \tag{5-6}$$

$$F_{out} = Norm\big(concat(F_1, F_2, F_3, F_4)\big) \tag{5-7}$$

5.2.3 基于余弦距离的 K 邻近算法

自 PointNet++提出以来，点云学习的层次化结构基本被确定为采样、分组、特征提取三个步骤。PointNet++在分组环节使用 K 邻近算法查询距离某个点最近的 K 个点，这种查询实质上是在三维空间中计算三维点的距离，并截取最近的 K 个点的坐标为结果。DGCNN 提出了点云的动态图卷积思想，其核心原理在于通过 K 邻近算法在特征空间中对特征进行邻近查询，本质上是在欧几里得空间中计算特征与特征之间的距离，并将距离最近的 K 个特征作为结果。可以看出上述两种计算方式都是在计算欧几里得距离（Euclidean Distance），只是前者计算坐标的欧几里得距离，而后者计算特征的距离。欧几里得距离的计算公式如式(5-8)所示。

$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{5-8}$$

特征相似度有许多计算方法，其中包括欧几里得距离、曼哈顿距离、余弦距离等。余弦距离是余弦相似度的另一种表示，余弦相似度计算公式如(5-9)所示。

$$T(x,y)=\frac{\sum x_i y_i}{\sqrt{\sum {x_i}^2}\sqrt{\sum {y_i}^2}} \tag{5-9}$$

$$d_{cos}(x,y)=1-T(x,y) \tag{5-10}$$

余弦距离如式(5-10)所示，由公式可得余弦距离的范围是$[0,2]$。作为距离表示，余弦距离数值越小代表两个向量越接近。KNN 算法作为一种相似度衡量方法，根据其参考距离的不同已有多种尝试，其中基于余弦距离的 KNN 算法[81-82]常常在文本分类问题或定位问题中应用。欧几里得距离衡量了两个向量之间的相似度，同样的余弦相似度也可以衡量两个向量之间的相似度，余弦距离可以衡量两个向量在相似空间中的距离。此外，余弦相似度可以将向量之间的相对位置计算得出，而欧几里得距离无法规避向量中极端数值的影响。如图 5-4 所示，以二维平面为例，p_1、p_2和q为平面的三个向量，夹角为θ，因此$T(p_1,q)$和$T(p_1,q)$相等。图 5-4 中虚线为p_1和q的欧氏距离，实线为p_2和q的欧氏距离，可以看到前者要小于后者，。

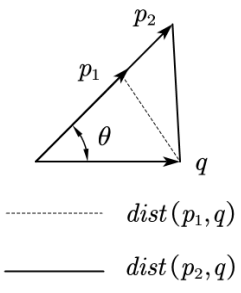

图 5-4 余弦距离与欧氏距离对比

受 PointNet++分组思想和 DGCNN 的特征分组思想的启发，本节网络将点云特征以余弦距离作为参考，通过余弦距离计算特征之间的相似性，并取最相似的 K 个特征作为邻近特征，加强特征提取的语义预测。实施过程如算法 5-2 所示。

算法 5-2 基于余弦距离的 K 邻近算法

算法：基于余弦距离的 K 邻近算法
输入：点云关键点数据 B x D x S，整体点云数据 B x D x N，B 为批次大小，D 为特征维度，S 为关键点数量，N 为整体点云数量
参数：邻近点数量 K
输出：邻近点数据 B x D x S x K
算法流程： 1. 计算点云关键点数据与整体点云数据之间的特征维度 D 上的余弦相似度，得到余弦相似度矩阵 B x S x N 2. 将余弦相似度矩阵转换为余弦距离矩阵，计算每个关键点的最小的 K 个余弦距离，并得到 K 个最小余弦距离所在的矩阵索引 3. 通过矩阵索引在整体点云数据中查询得到所求的 K 个邻近点的信息 B x D x S x K

5.3 基于通道自注意力的点云分割网络

5.3.1 网络架构设计

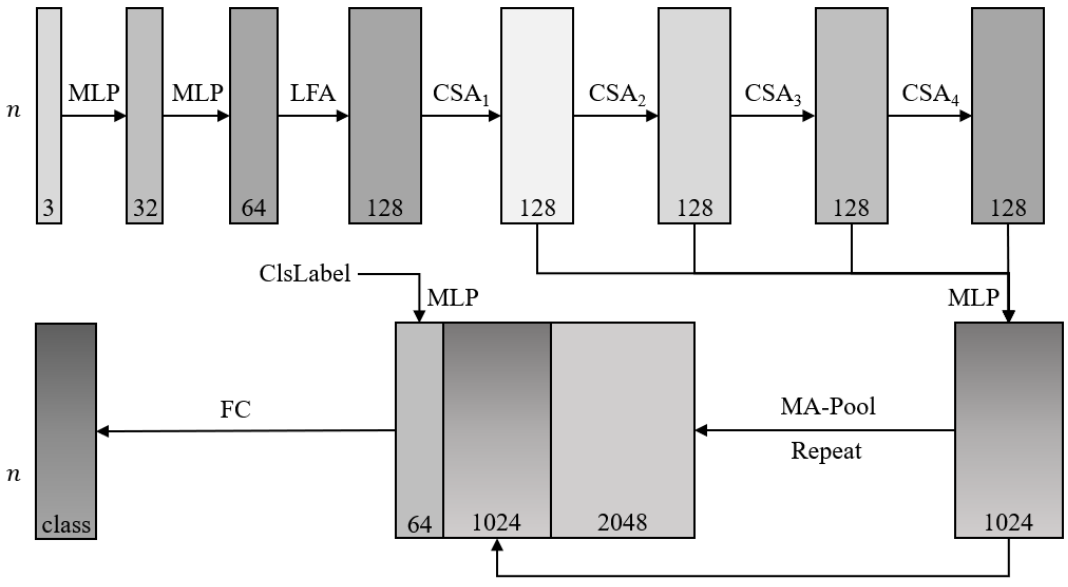

图 5-5 基于通道自注意力的点云分割网络架构

本节根据前文介绍的模块和 DGCNN 主干来构建三维点云分割网络。网络架构如图 5-5 所示，网络的编码器与上一节分类网络相似，编码器由一层局部特征抽象模块（LFA）和一层四头通道自注意力（CSA）模块组成，两种模块直接串联。

在分割任务中，由于网络最终输出需为每个点的预测结果，因此 LFA 模块并不对点云下采样，而是直接对输入的点云进行特征学习。LFA 模块后紧跟一个四头通道自注意力模块，对所有的点特征进行语义学习和修正。网络解码器不同于分类网络，四头通道注意力模块输出的 1024 维特征向量首先进行最大池化和平均池化，两种池化得到的向量重复后与原有特征向量拼接，得到 3072 的特征向量，然后将分割类别 One-Hot 向量同样拼接上特征向量，得到的点云分割特征最后通过多个全连接层得到预测结果，全连接层之间包括 BatchNorm 和 Dropout。

基于通道自注意力的点云分割网络的详细参数如表 5-1 所示，点云输入为 2048×3，代表 2048 个点和对应的空间位置坐标。随后每一层直接对 2048 点进行特征学习，不进行下采样操作。点云输入首先通过两层多层感知机，将输入信息扩展到特征空间，随后 LFA 层对 2048 点的特征应用基于余弦距离的 K 邻近查询，得到 32 个点特征，最后通过池化来抽象为关键点的特征。表中 4-CSA 代表四头通道自注意力机制，每个头的输入为 128 维，输出为 128 维，将四个头的输出拼接在一起作为下一层的输入。解码器由三层全连接层组成，包括 BatchNorm 和 ReLU。每层全连接层输出维度分别为 512、256、50，其中 50 是部分分割的总类别数量。

表 5-1 网络模块参数

Module	N	K	InputChannel	OutputChannel
MLP1	2048	-	3	32
MLP2	2048	-	32	64
LFA	2048	32	70	128
4-CSA	2048	-	128	512
MLP3	2048	-	512	1024
FC1	2048	-	3136	512
FC2	2048	-	512	256
FC3	2048		256	50

5.3.2 定性评估与定量分析

本章节根据前文介绍的模块和架构来建立三维点云分割网络，实现本章所介绍的基于层次化注意力的点云分割网络。网络在 ShapeNet 数据集上进行实验和评估，采用 NVIDIA RTX 2080Ti 进行加速训练，详细实验环境见表 5-2 所示。

表 5-2 实验环境

机器编号	系统平台	CPU	GPU	显存	RAM	Cuda
1	Ubuntu 18.04	i7-8700	RTX 2080Ti	11GB	16GB	10.1

ShapeNet 数据集设置与 DGCNN 设置相同，输入数据是在网格模型中均匀采样的 2048 个点。训练总轮次 Epoch 设置为 200，batchsize 设置为 16，测试 batchsize 设置为 8，在训练过程中采用随机梯度下降优化器（SGD）来更新学习率，初始学习率设置为 0.001，衰减系数设置为 0.0001。

本节采用第四章提到的两个评价指标：平均实例交并比（Ins.IoU）和平均类别交并比（Cls.IoU），将网络的实验结果与其他优秀网络进行定量对比和分析，实验对比结果如表 5-3 所示。

表 5-3 部分分割实验结果

Method	PointNet[33]	Kd-Net[25]	PointNet++[34]	TransformerNet	DGCNN[38]	CANet
Ins.IoU	83.7	82.3	85.1	84.2	85.2	**85.9**
Cls.IoU	80.4	-	81.9	80.6	82.3	**83.1**
airplane	83.4	80.1	82.4	82.5	84.0	**85.2**
bag	78.7	74.6	79.0	75.6	83.4	**84.1**
cap	82.5	74.3	87.7	83.1	86.7	**87.9**
car	74.9	70.3	77.3	76.0	77.8	**79.7**
chair	89.6	88.6	90.8	90.2	90.6	**91.4**
earphone	73.0	73.5	71.8	72.9	**74.7**	71.3
guitar	91.5	90.2	91.0	91.2	91.2	**92.0**
knife	85.9	87.2	85.9	87.9	87.5	**88.1**
lamp	80.8	81.0	83.7	83.5	82.8	**84.0**
laptop	95.3	94.9	95.3	95.6	95.7	96.1
motorbike	65.2	57.4	**71.6**	64.3	66.3	71.3
mug	93.0	86.7	94.1	94.0	**94.9**	94.7
pistol	81.2	78.1	81.3	80.6	81.1	**84.1**
rocket	57.9	51.8	58.7	59.1	**63.5**	61.0
skateboard	72.8	69.9	**76.4**	72.0	74.5	76.0
table	80.6	80.3	82.6	81.6	**82.6**	82.5

表 5-3 中，将本章网络与 PointNet、PointNet++、DGCNN 等网络进行比较，分析表中结果数据可知，本章网络 CANet 分别在 Ins.IoU 和 Cls.IoU 上取得了 85.9%和 83.1%优异成绩，并且在飞机、台灯、帽子、汽车、椅子等类别上均取得较好的成绩。相比于 DGCNN 网络，CANet 网络的 Cls.IoU 提高了 0.8%，这是因为 DGCNN 在点与点的关联特征上进行了卷积操作提取特征，但特征之间的相似性并没有充分利用，CANet 在其基础上不仅通过余弦距离在特征空间查询相似特征，而且通过注意力机制进一步提取相似特征的语义信息。相比于 PointNet++和第四章所提

出的网络 TransformerNet，本章网络在两项评价指标上都有着明显的提升，首先是由于通道自注意力模块引入了点云位置信息编码，并优化了注意力计算方式，因此网络在部分分割任务上较第四章基于 Transformer 的点云学习网络有较大的提升。其次针对第四章邻近点过多而污染语义空间的问题，本章算法采用余弦距离控制关键点与邻近点的语义相似性，尽可能避免了上述问题的发生，提高了网络的拟合能力。综合对比分析，本章提出的基于通道自注意力的点云分割网络可以充分学习点云各部分的语义信息，具有较强的优越性。

除综合对比分析外，本节对 KNN 的距离参考值进行了对比实验，探讨分割实验中距离因素对 KNN 算法的影响。如表 5-4 所示，以欧几里得距离和余弦距离作为对比，由于点云数据的原始三维坐标并不包含明显的语义相似信息，而三维空间中的相邻点有相近的语义的可能性更大，因此本次实验不考虑计算点云坐标的余弦距离。表中 Dist(xyz)代表计算点云三维坐标的欧氏距离，Dist(feature)代表计算点云特征向量之间的欧氏距离，而 Consine(feature)代表计算特征之间的余弦距离。由表 5-4 数据可知基于余弦距离的 KNN 算法取得了 85.9%的由优异表现，与欧氏距离相比，余弦距离作为点云特征的语义相似度算法更能实现语义的精准聚类，也增强了点云网络的精准分割能力。

表 5-4 不同的 KNN 参考值对比实验

算法	KNN 值	Ins.IoU
CANet	Dist(xyz)	85.5
CANet	Dist(feature)	85.7
CANet	Cosine(feature)	85.9

对本章网络进行可视化定性评估，如图 5-6 所示，图中将本章网络在不同 KNN 参考距离下的模型结果进行可视化，并与数据集的真实标签相比较。由图 5-6 可以看出，本章提出的网络具有较强的特征提取能力，并且通过定性评估也证明了基于余弦距离的 K 邻近算法的有效性，图中较为明显的是对桌子的分割，其中参考余弦距离的 KNN 算法效果最佳，参考欧氏特征距离的次之。由于点云特征提取的核心仍旧是对称函数，而现有的方法最常用的对称函数是最大池化函数，因此点云在进行采样和分组时，期望邻近点语义与关键点相似，若邻近点语义与关键点相差较大，则关键点的语义信息就有可能被错误预测。本节的定量分析和定性评估展示了基于余弦距离的 K 邻近算法在查询点云语义相似点的强大能力，也论证了对特征进行语义相似度查询要比对坐标查询更加有效，这是因为坐标在未经过网络学习时，所表征的信息有限，三维空间中邻近的位置信息不代表在语义空间中相似。此

外，本章网络在语义空间中通过注意力机制对语义进行了提取和修正，使得语义逐渐准确，与本章采用的基于余弦距离的 K 邻近算法相辅相成，最终展示了在 ShapeNet 数据集上的优异表现。

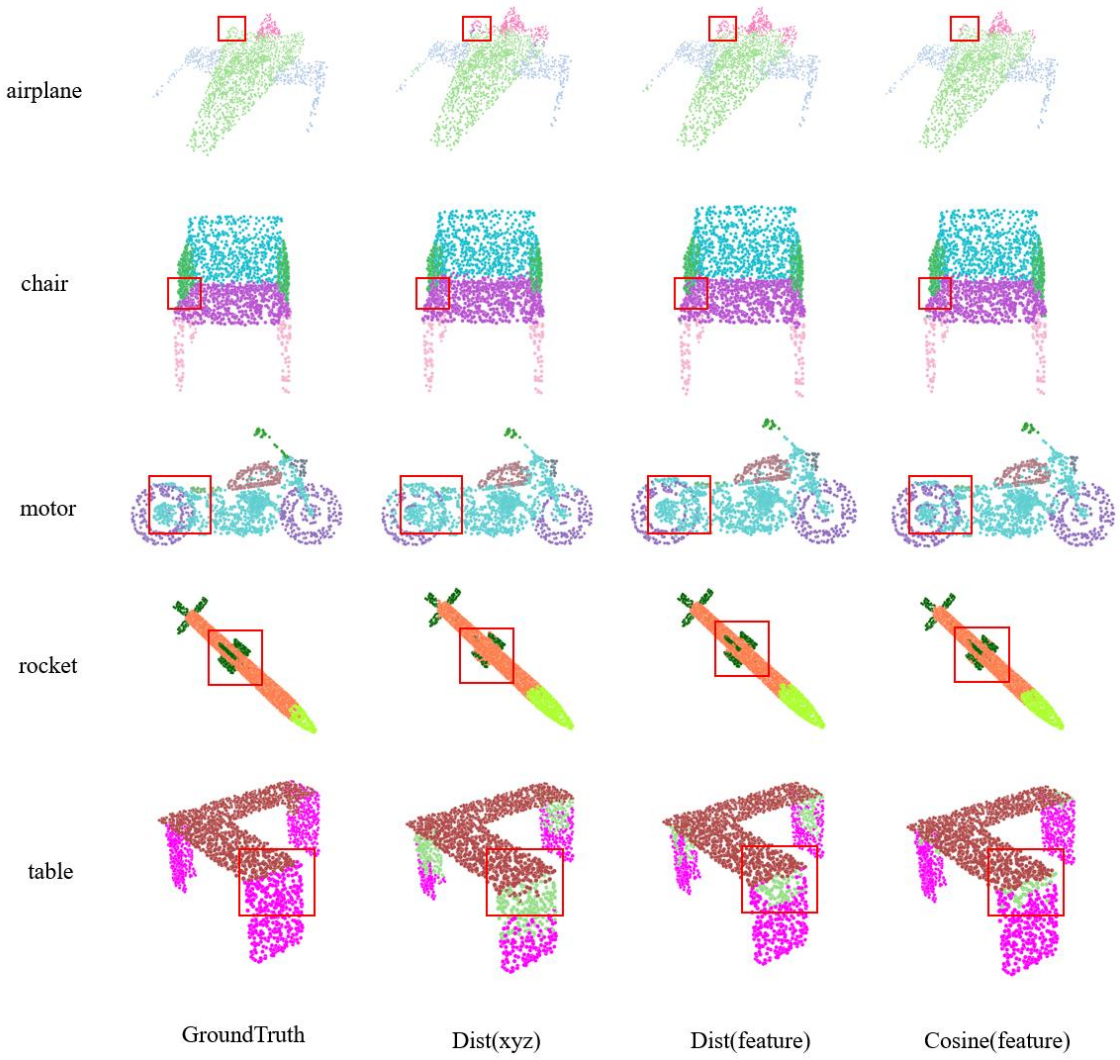

图 5-6 KNN 参考值对比实验可视化

5.3.3 本节网络与层次化结构比较

本章基于通道自注意力的点云分割网络以 DGCNN 为主干，并没有使用采样分组思想，本节中，根据 PointNet++结构搭建一种层次化的点云分割网络，并将本章设计的模块嵌入到该网络中进行实验评估，最终和本章分割网络进行对比分析。本节层次化分割网络（Hierarchical Attention Net, HANet）以 PointNet++为基础主干框架来搭建，并采用自注意力机制（Self-Attention, SA）。网络整体设计如图 5-7 所示，HANet 由编码器和解码器组成，编码器包括局部特征抽象模块、通道自注意

力模块组成。解码器由两个特征传播模块（FeaturePropagation, FP）和一个通道自注意力模块组成。

图 5-7 HANet 结构

在分割任务中，网络需要对输入点云的每个点进行预测，而点云经过局部特征抽象模块后已经被采样分组，点云数量相应下降，为解决该问题，分割任务的解码器采用 PointNet++中的特征传播模块，将采样后点云的特征通过插值算法传播到原始点云上。此外，为了解决插值时特征准确率较低的问题，本章网络在两层特征传播模块之间加入通道自注意力模块，辅助特征传播模块进行特征插值。

表 5-5 网络模块参数

Module	N	K	InputChannel	OutputChannel
LFA1	512	32	3	64
4-CSA1	512	-	64	128
LFA 2	256	64	128	256
4-CSA2	256	-	256	512
MLP	256	-	512	1024
FP1	256	-	768	[256,128]
4-CSA3	256	-	128	128
FP2	512	-	150	[128,128,128]
FC2	2048	-	256	40

网络详细参数如表 5-5 所示。编码器与分类结构一致，每层局部特征抽象模块的点数与特征输入输出维度分别为$(512, 64)$、$(256, 256)$。编码器中通道自注意力模块的输出维度为 256、512。表中 4-CSA 代表四头通道自注意力机制。分割结构中将编码器得到的每层点云坐标和特征通过 FP 模块进行点云的上采样，并将局部特征插值到全局点云中，特征传播层的维度为 768、150。

本节层次化分割网络 HANet 的实验环境和实验设置与 CANet 一致，网络在 ShapeNet 数据集上进行实验和评估，实验结果如表 5-6 所示。本节网络 HANet 与第四章所提出的网络对比，分割 mIoU 高出了 0.4%，HANet 中的通道自注意模块和局部特征抽象模块提高了分割网络的特征学习能力，并且层次化结构对每个点的分割任务有着天然的适配性，因为对局部特征有良好的提取也就意味着对每个

点的语义有着精准的预测。然而，HANet 与本章提出的 CANet 相差 1.3%，这是因为 HANet 对点云进行了下采样，输入点数量为 2048 的点云采样到 512 个点，损失了一定的点云信息。

表 5-6 层次化分割网络对比实验

Method	PointNet[33]	PointNet++[34]	TransformerNet	CANet	HANet
Ins.IoU	83.7	85.1	84.2	**85.9**	84.6
Cls.IoU	80.4	81.9	80.6	**83.1**	80.8
airplane	83.4	82.4	82.5	**85.2**	82.5
bag	78.7	79.0	75.6	**84.1**	80.9
cap	82.5	87.7	83.1	**87.9**	83.6
car	74.9	77.3	76.0	**79.7**	76.2
chair	89.6	90.8	90.2	**91.4**	90.2
earphone	**73.0**	71.8	72.9	71.3	72.6
guitar	91.5	91.0	91.2	**92.0**	90.9
knife	85.9	85.9	87.9	**88.1**	87.7
lamp	80.8	83.7	83.5	**84.0**	83.5
laptop	95.3	95.3	95.6	**96.1**	95.2
motorbike	65.2	**71.6**	64.3	71.3	57.3
mug	93.0	94.1	94.0	**94.7**	94.3
pistol	81.2	81.3	80.6	**84.1**	81.2
rocket	57.9	58.7	59.1	61.0	**61.1**
skateboard	72.8	**76.4**	72.0	76.0	73.0
table	80.6	82.6	81.6	82.5	**82.7**

本节网络 HANet 在分割表现上优于 TransformerNet，劣于 CANet，然而在模型大小和计算资源的需求上较为轻量，如表 5-7 所示。与 TransformerNet 相比，HANet 在算力占劣势的 2080ti 上也有着较快的训练时间，参数量也少于其他两种网络。与 CANet 相比，本节网络 HANet 十分轻量，这得益于层次化结构对点云的采样操作，但同时也降低了网络的精准分割能力。

表 5-7 训练时间对比

算法	设备	点云数量	批次大小	参数量	训练时间(秒/epoch)	Ins.IoU
TransformerNet	3090	2048	32	0.84M	220	84.2
HANet	2080ti	2048	32	0.64M	98	84.6
CANet	2080ti	2048	16	2.64M	384	85.9

从表 5-7 的实验数据可以看出 CANet 在时间效率方面要低于第四章 TransformerNet。然而 TransformerNet 在 3090 显卡上完成实验，并且批次大小也是 CANet 的两倍，3090 显卡在显存和算力上均强于 2080ti 显卡。因此 CANet 网络的运行速度并不弱于 TransformerNet。经过研究分析发现，CANet 中的通道自注意力模块使用了较高的输出维度，导致卷积操作的参数量变大，更多的参数和矩阵运算导致神经网络整体的参数量和计算时间呈倍数增长。

对本节网络进行可视化定性评估，如图 5-8 所示，图中将第四章网络 TransformerNet 和本章所提出的 HANet、CANet 进行可视化结果分析。由图 5-8 可以看出，CANet 和 HANet 的效果比较明显，并且在细节方面 CANet 分割更加精准。在部分与部分之间的连接处，网络难以区分部分的类别，例如对手提包进行分割时，提手和包体往往容易混淆，同样的水杯中的杯身和杯手的预测也有误差。在上述情况下，本章提出的两个网络可以有效区分部分关联处的点云，并且 CANet 较 HANet 有更精准的分割能力。此外，由于 HANet 对点云进行下采样，输入的点云数量少于 CANet，因此在细节上和全局上都与 CANet 有着一定差距，例如对耳机和帽子的分割，尤其是帽子的分割在全局上有所误差，帽檐分割效果较好，但帽子顶部有一部分被预测为帽檐。

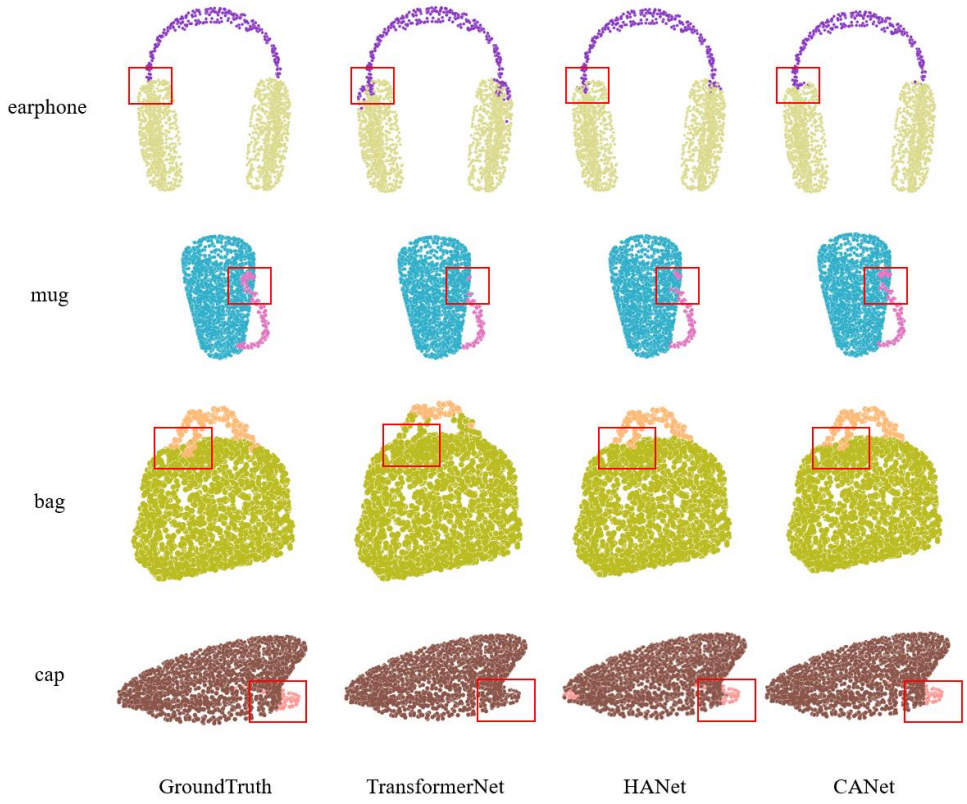

图 5-8 部分分割可视化对比

5.4 小结

本章的内容主要分为三个部分。

第一部分讲述了本章算法解决的问题和应用的主要模块，介绍了本章的基本内容和主要贡献，对局部特征抽象模块和基于余弦距离的 K 邻近算法进行了简述。

第二部分针对点云分割任务，设计了本章算法的具体结构和方法，首先基于采样分组思想设计了一种局部特征抽象模块，不仅对特征进行了相似性聚类，也将点云位置信息编码入局部特征中。局部特征抽象模块巧妙的将余弦相似度应用于 K 邻近算法中，通过余弦距离来衡量特征向量的相似性。其次在自注意力机制中优化计算效率，设计了一种基于通道自注意力的特征提取模块，保留注意力思想的同时加快计算速度。

第三部分利用本章设计的算法和模块，搭建了基于通道自注意力的点云分割网络，并在公开数据集上进行了实验和分析。此外，在采样分组的思想上提出了基于层次化的点云分割网络，并于本章网络进行定量实验和定性评估，通过大量实验证明了本章网络在点云部分分割上的优势，并且对网络参数和计算时间进行对比分析，将网络分割结果进行可视化，进一步验证了本章网络和层次网络在精准分割和计算效率上的优势，以及本章算法在点云分割领域中的优越性和可靠性。

第六章 总结与展望

6.1 全文总结

点云数据在计算机视觉中愈发重要，本文针对点云语义分割领域中的场景数据利用不充分和关联特征的提取问题，提出了融合 RGB 特征的点云分割网络、基于 Transformer 的点云学习网络和基于通道自注意力的点云分割网络，基本思路是通过引进新的特征数据和特征提取机制，提高点云网络对语义的学习能力。本文的主要工作和贡献如下：

（1）第一部分针对场景数据的利用不完全问题，设计了一种融合 RGB 特征的算法，该算法可以将点云数据与图像特征相融合，有效利用场景数据，并提出了一种融合 RGB 特征的点云分割网络，该网络优化了点云提取特征算法，提出了相对特征提取模块和点云交叉空间注意力模块进行点云特征提取。实验证明本章融合算法的有效性和网络在场景语义分割任务中的精准。

（2）第二部分针对点云的关联性特征提取问题，提出了基于 Transformer 的点云学习网络，引入了特征位置编码，通过计算特征空间的特征距离来计算特征间相似性，并优化自注意力机制的结构和正则化方式，使网络既能对点云关联特征进行学习，也可以对局部特征进行语义修正。将优化后的网络在点云分类、部分分割、场景语义分割进行了综合实验，与其它优秀的算法进行分析和对比，证明本章算法具有较强的特征学习能力和泛化能力。

（3）第三部分针对注意力机制的计算效率问题，本章对自注意力进行优化改进，提出了基于通道自注意力的点云分割网络，使得计算效率和网络性能得到提高。针对点云特征的相似性问题，本章将基于余弦距离的 K 邻近算法应用于点云数据，采用余弦距离作为 K 邻近算法参考距离，余弦相似度用来衡量特征之间的相似性。引入层次化思想，进一步提高网络学习能力，精简网络参数。设计本章邻近算法的对比实验和本章网络的综合实验，统计了网络的精度和性能，检验并证明了本章算法具有较为轻量的结构和优秀的场景分割能力。

6.2 工作展望

本文在点云语义分割领域对融合数据和注意力的方法进行了分析和研究，在分类任务和分割任务中取得了良好的效果，解决了点云语义分割的一些问题和挑

战。虽然本文研究取得了一定的效果，但同时仍具有一定的不足和可以提升的空间，对未来相关研究工作展望如下：

（1）在本文提出的融合 RGB 特征的点云分割网络和基于 Transformer 的点云学习网络中，由于网络集成了多种模块，结构较为复杂，网络的输入或层数较多，导致网络参数量较大，模型训练时间较长。其中融合 RGB 特征的点云分割网络采用二维网络和三维网络，模型复杂度大大增加，并且融合方法依赖图像特征的提取，在精度和效率上都会受到二维图像特征的影响。基于 Transformer 的点云学习网络由于分割任务中点云数量庞大，并且需对每点的语义进行预测，网络结构难以压缩。

（2）在本文提出了基于 Transformer 的点云学习网络和基于通道自注意力的点云分割网络中，网络采用了自注意力机制对点云进行提取，与先前方法对比有了较大的进步，然而和现实应用所要求的精度相比还有一定的差距。基于通道自注意力的点云分割网络虽然对当前的模型结构和参数进行了优化，减少了参数量和训练时间，但仍然受限于显存等物理设备条件，无法进行在线语义分割，这也是未来可以研究的方向。

致　谢

七年时间转瞬即逝，我的学生生涯马上要落下帷幕，开启新的篇章。在成电的生活，有欢喜也有焦虑，所有的一切都是命运带给我的礼物。在研究生的最后阶段，本文工作即将完成之际，我想对关心、帮助过我的人真诚地说一声谢谢，正是有你们在，人生的这一阶段才有了丰富多彩的意义。

首先感谢我的导师——饶云波老师，老师不仅在科研上给予了我很大的指导，在生活上也给予了很多关心。不管是治学，还是生活，无论是方法还是态度，我都从老师身上学习到很多，这对我以后的工作、学习和生活有着深远的影响。

其次我要感谢实验室的所有同学，感谢各位同学提供的各种指导和经验分享，通过在实验室探讨学习，让我的技术能力有了长远的进步。同时也感谢身边各位同学在生活上的陪伴，让我在快乐的科研氛围中完成了研究生学业。

最后感谢我的家人，感谢在求学路上的关心和鼓励，家人是我永远的后盾，正是有了家人的支持，我才能在人生的路上不断前进，是他们给了我人生的力量，我也会一路向前，努力去爱家人，去爱身边每一个人。

参考文献

[1] 王玉婷. RGB-D 图像和点云图像实例分割方法研究[D]. 合肥工业大学, 2019.

[2] 李静. 基于卷积神经网络的三维模型分割算法研究与实现[D]. 中北大学, 2020.

[3] 周佳新. 基于语义信息的图像/点云配准与三维重建[D].中国科学院大学(中国科学院深圳先进技术研究院),2020.

[4] 李博杨. 基于三维激光雷达的道路环境感知[D].北京交通大学,2019.

[5] 邹遇,熊禾根,陶永,等.基于 3 维点云深度信息和质心距相结合的机器人抓取控制方法[J].高技术通讯,2020(5):508-517.

[6] 刘立恒,赵夫群,汤慧,等. 几何特征保持的文物点云去噪算法[J]. 数据采集与处理,2020,35(2):373-380. DOI:10.16337/j.1004-9037.2020.02.019.

[7] 盛添宇. 基于深度传感器的虚拟呈现技术研究[D].南京邮电大学,2019.

[8] 廖书航.基于视觉 SLAM 稀疏点云的增强现实应用[J].现代计算机(专业版),2019(05):56-59.

[9] 汤怡君. 大范围室外场景三维点云语义分割[D]. 大连理工大学, 2019.

[10] 刘念. 基于视觉的多机器人 SLAM 算法研究与实现[D].电子科技大学,2020.

[11] 翟少华. 基于图像和点云融合的道路障碍物感知与参数化分析[D].哈尔滨工业大学,2020.

[12] 王柯,易琳,钱金菊,等.一种高效的机载激光雷达点云电力线塔全自动分割方法[J].地理信息世界,2020,27(04):115-118+122.

[13] 孙月霞,李梁杭,朱铃铃.基于点云数据的文物修复与建模[J].测绘与空间地理信息,2020,43(10):57-59.

[14] 肖震. 基于八叉搜索的点云语义分割网络设计与实现[D]. 大连理工大学, 2020.

[15] 张灿. 基于卷积神经网络的图像语义分割算法研究[D].华中科技大学,2017.

[16] Guo Y, Wang H, Hu Q 等. Deep learning for 3D point clouds: a survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(12): 4338 - 4364. https://ieeexplore.ieee.org/document/9127813/. DOI:10.1109/TPAMI.2020.3005434.

[17] 张佳颖,赵晓丽,陈正.基于深度学习的点云语义分割综述[J].激光与光电子学进展,2020,57(04):28-46.

[18] 顾军华,李炜,董永峰.基于点云数据的分割方法综述[J].燕山大学学报,2020,44(02):125-137.

[19] 景川. 基于深度学习的三维点云语义分割研究[D]. 西安电子科技大学, 2019.

[20] 党吉圣, 杨军. 多特征融合的三维模型识别与分割[J]. 西安电子科技大学学报, 2020, 47(4):9.

[21] Maturana D, Scherer S. Voxnet: a 3d convolutional neural network for real-time object recognition[C]. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2015: 922-928.

[22] Wu Z, Song S, Khosla A, et al. 3D Shapenets: a deep representation for volumetric shapes[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1912-1920.

[23] Li Y, Pirk S, Su H, et al. FPNN: Field probing neural networks for 3d data[J]. Advances in Neural Information Processing Systems, 2016, 29.

[24] Riegler G, Osman Ulusoy A, Geiger A. Octnet: learning deep 3d representations at high resolutions[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 3577-3586.

[25] Klokov R, Lempitsky V. Escape from cells: Deep kd-networks for the recognition of 3d point cloud models[C]. Proceedings of the IEEE International Conference on Computer Vision. 2017: 863-872.

[26] Tatarchenko M, Dosovitskiy A, Brox T. Octree generating networks: efficient convolutional architectures for high-resolution 3d outputs[C]. Proceedings of the IEEE International Conference on Computer Vision. 2017: 2088-2096.

[27] Wang P S, Liu Y, Guo Y X, et al. O-cnn: Octree-based convolutional neural networks for 3d shape analysis[J]. ACM Transactions On Graphics (TOG), 2017, 36(4): 1-11.

[28] Le T, Duan Y. Pointgrid: a deep network for 3d shape understanding[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 9204-9214.

[29] 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.

[30] Qi C R, Su H, Nießner M, et al. Volumetric and multi-view cnns for object classification on 3d data[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 5648-5656.

[31] Lawin F J, Danelljan M, Tosteberg P, et al. Deep projective 3d semantic segmentation[C]. International Conference on Computer Analysis of Images and Patterns. Springer, Cham, 2017: 95-107.

[32] Kundu A, Yin X, Fathi A, et al. Virtual multi-view fusion for 3d semantic segmentation[C]. European Conference on Computer Vision. Springer, Cham, 2020: 518-535.

[33] 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.

[34] 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.

[35] Zhang Z, Hua B S, Yeung S K. Shellnet: efficient point cloud convolutional neural networks using concentric shells statistics[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 1607-1616.

[36] Hu Q, Yang B, Xie L, et al. Randla-net: efficient semantic segmentation of large-scale point clouds[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 11108-11117.

[37] Landrieu L, Simonovsky M. Large-scale point cloud semantic segmentation with superpoint graphs[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 4558-4567.

[38] 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.

[39] Thomas H, Qi C R, Deschaud J E, et al. Kpconv: Flexible and deformable convolution for point clouds[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 6411-6420.

[40] Hua B S, Tran M K, Yeung S K. Pointwise convolutional neural networks[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 984-993.

[41] Huang Q, Wang W, Neumann U. Recurrent slice networks for 3d segmentation of point clouds[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 2626-2635.

[42] Jiang M, Wu Y, Zhao T, et al. Pointsift: a sift-like network module for 3d point cloud semantic segmentation[J]. arXiv preprint arXiv:1807.00652, 2018.

[43] Li Y, Bu R, Sun M. PointCNN: convolution on X-transformed points[J]. Advances in Neural Information Processing Systems, 2018, 2018-Decem: 820–830.

[44] Zhao H , Jiang L , Fu C W , et al. PointWeb: enhancing Local Neighborhood Features for Point Cloud Processing[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019: 5565-5573.

[45] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems, 2012, 25.

[46] 肖建桥. 基于深度学习的道路场景语义分割[D]. 吉林大学, 2019.

[47] Simonyan K , Zisserman A . Very Deep convolutional networks for large-scale image recognition[J] . 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, 2015: 1–14.

[48] 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.

[49] Dai A, Niessner M. 3DMV: joint 3D-multi-view prediction for 3d semantic scene segmentation[C]. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). . DOI:10.1007/978-3-030-01249-6_28.

[50] Liang M, Yang B, Wang S , et al. Deep continuous fusion for multi-sensor 3d object detection[J]. Lecture Notes in Computer Science, 2018, 11220 LNCS: 663–678. DOI:10.1007/978-3-030-01270-0_39.

[51] Jaritz M, Gu J, Su H. Multi-view pointnet for 3D scene understanding[C]//Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019. IEEE, 2019: 3995–4003. DOI:10.1109/ICCVW.2019.00494.

[52] Sun Y, Zuo W, Yun P, et al. FuseSeg: semantic segmentation of urban scenes based on RGB and thermal data fusion[J]. IEEE Transactions on Automation Science and Engineering, 2020, 18(3): 1000-1011.

[53] Chen L Z, Li X Y, Fan D P, et al. LSANet: Feature learning on point sets by local spatial aware layer[J]. arXiv preprint arXiv:1905.05442, 2019.

[54] Zhao C, Zhou W, Lu L, et al. Pooling scores of neighboring points for improved 3D point cloud segmentation[C]. 2019 IEEE International Conference on Image Processing (ICIP). IEEE, 2019: 1475-1479.

[55] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30.

[56] 张晓旭,马志强,刘志强等.Transformer 在语音识别任务中的研究现状与展望[J].计算机科学与探索,2021,15(09):1578-1594.

[57] 任欢,王旭光.注意力机制综述[J].计算机应用,2021,41(S1):1-6.

[58] Yin W, Kann K, Yu M, et al. Comparative study of CNN and RNN for natural language processing[J]. arXiv preprint arXiv:1702.01923, 2017.

[59] 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.

[60] Wang Y, Xu Z, Wang X, et al. End-to-end video instance segmentation with transformers[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 8741-8750.

[61] Zhao H, Jiang L, Jia J, et al. Point transformer[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 16259-16268.

[62] Guo M H , Cai J X , Liu Z N , et al. PCT: point cloud transformer[J]. Computational Visual Media, 2021, 7(2): 187–199. DOI:10.1007/s41095-021-0229-5.

[63] Yu J, Zhang C, Wang H, et al. 3D medical point transformer: introducing convolution to attention networks for medical point cloud analysis[J]. arXiv preprint arXiv:2112.04863, 2021.

[64] Xu G, Cao H, Wan J, et al. Adaptive channel encoding transformer for point cloud analysis[J]. arXiv preprint arXiv:2112.02507, 2021.

[65] Pan X, Xia Z, Song S, et al. 3D object detection with pointformer[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 7463-7472.

[66] 余方洁,王斌. 基于 RGB-D 图像的移动端点云分割方法研究[J]. 重庆理工大学学报(自然科学), 2022, 36(2): 126-134.

[67] 李彤. 基于特征融合的三维点云场景理解研究及系统实现[D]. 北京邮电大学, 2021.

[68] Zaremba W, Sutskever I, Vinyals O. Recurrent neural network regularization[J]. arXiv preprint arXiv:1409.2329, 2014.

[69] Shi X, Chen Z, Wang H, et al. Convolutional LSTM network: a machine learning approach for precipitation nowcasting[J]. Advances in Neural Information Processing Systems, 2015, 28.

[70] Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation[C]. International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015: 234-241.

[71] Dai A, Chang A X, Savva M, et al. Scannet: richly-annotated 3d reconstructions of indoor scenes[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 5828-5839.

[72] 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.

[73] Yi L, Kim V G, Ceylan D, et al. A scalable active framework for region annotation in 3d shape collections[J]. ACM Transactions on Graphics (ToG), 2016, 35(6): 1-12.

[74] Armeni I, Sener O, Zamir A R, et al. 3D semantic parsing of large-scale indoor spaces[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1534-1543.

[75] Xie S, Liu S, Chen Z, et al. Attentional shapecontextnet for point cloud recognition[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 4606-4615.

[76] Han X F , Jin Y F , Cheng H X , et al. Dual transformer for point cloud analysis[J]. http://arxiv.org/abs/2104.13044, 2021.

[77] Yu X, Tang L, Rao Y, et al. Point-BERT: pre-training 3D point cloud transformers with masked point modeling[J/OL]. http://arxiv.org/abs/2111.14819, 2021.

[78] Ye X, Li J, Huang H, et al. 3D recurrent neural networks with context fusion for point cloud semantic segmentation[C]. Proceedings of the European Conference on Computer Vision (ECCV). 2018: 403-417.

[79] 谭锦钢. 基于 RGBD 的三维场景语义解析算法研究[D]. 中国科学院大学,2021.

[80] Abeywickrama T, Cheema M A, Taniar D. K-nearest neighbors on road networks: a journey in experimentation and in-memory implementation[J]. arXiv preprint arXiv:1601.01549, 2016.

[81] 黄运稳,陈光,叶建芳. 基于余弦相似度的加权 K 近邻室内定位算法[J]. 计算机应用与软件, 2019, 36(2): 159-162.

[82] 杜太行,孟岩,孙曙光,等. 基于改进加权 KNN 算法的室内无线电发射源定位研究[J]. 中国测试, 2019, 45(9): 105-111.

攻读硕士学位期间取得的成果

- **发表的论文**

[1] Rao Y, Mu H, **Yang Z** et al, B-pesnet: Smoothly propagating semantics for robust and reliable multi-scale object detection for secure systems[J]. Computer Modeling in Enginerring & Sciences, 2022.

- **申请专利**

[1] 饶云波，**杨泽宇**等，一种融合图像特征的三维点云场景分割方法及系统[P]. 中国，发明专利. 202111423794.9, 2021-11-27.

- **参与项目情况**

[1] 面向云制造+边缘制造的工业互联网平台关键技术，项目编号 NO: 2020YFG0459

[2] 面向数据驱动的大规模场景分割与模型修复，项目编号 NO: 2021YFG0314

- **在校期间获得荣誉**

[1] 2020-2021 学年获学校研究生二等奖学金

[2] 2019-2020 学年获学校研究生二等奖学金