深 圳 大 学

本 科 毕 业 论 文（设计）

题目：基于注意力机制的3D场景图生成算法研究

姓名：张子扬

专业：计算机科学与技术

学院（部）：计算机与软件学院

学号：2019112005

指导教师：王旭

职称：副教授

2023年 5 月 7 日

深圳大学本科毕业论文（设计）诚信声明

本人郑重声明：所呈交的毕业论文（设计），题目《基于注意力机制的 3D 场景图生成算法研究》是本人在指导教师的指导下，独立进行研究工作所取得的成果。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注明。除此之外，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。本人完全意识到本声明的法律结果。

毕业论文（设计）作者签名：张子瑜

日期：　　2023 年 5 月 7 日

目 录

摘要(关键词)....1

1 引言....2

1.1 研究背景及意义....2

1.2 国内外研究现状....3

1.3 本论文的主要工作....3

1.4 论文的组织架构....4

2 相关工作....4

2.1 3D 点云处理技术....4

2.2 图神经网络....4

2.3 注意力机制....5

3 基于注意力的场景图预测网络....5

3.1 问题描述....5

3.2 模型结构....6

3.2.1 点云特征提取....6

3.2.2 节点逐特征注意力....7

3.2.3 边注意力掩码....9

3.2.4 类别预测和损失计算....10

4 实验评估和可视化....10

4.1 实验环境....10

4.2 3DSSG 数据集....11

4.3 基线方法....13

4.4 数据预处理....14

4.5 评估指标....16

4.6 语义场景图预测....16

4.7 消融实验....19

5 总结与展望....20

5.1 优点与局限性....20

5.2 未来工作展望....20

参考文献 21
致谢 23
Abstract(Key words) 24

基于注意力机制的 3D 场景图生成算法研究

计算机与软件学院计算机科学与技术专业　张子扬

学号：2019112005

【摘要】场景图是一种结构化的场景表示方式，用于清晰地表达场景中的物体、物体的属性以及物体间的关系。与从 2D 图像生成场景图类似，3D 场景图生成算法目标是从 3D 场景中构建提供准确量化 3D 场景中对象关系的表示方式，对于理解室内复杂环境等任务非常有帮助，具有重大研究意义。场景图预测网络（Scene Graph Prediction Network，SGPN）利用图神经网络（Graph Neural Network，GNN）来学习场景图物体和关系之间特征信息，是目前最先进的 3D 场景图生成框架之一。由于 GNN 的传播策略难以捕捉复杂 3D 场景中的视觉线索，本文在 SGPN 算法的基础上，提出了基于注意力机制的 3D 场景图生成算法优化思路：1）对于节点特征，本文引入了一种逐特征的注意力机制来捕捉物体和关系之间的权重信息；2）对于关系特征，本文引入了一种利用节点特征的注意力机制学习关系特征。实验结果表明，本文所提出的优化模型在包含 1482 个场景的 3DSSG 数据集上的召回率比 SGPN 提升了 19.68%。更多消融实验和可视化展示进一步证实了优化后模型的有效性和鲁棒性。

【关键词】3D 场景理解；场景图生成算法；图神经网络；注意力机制

1 引言

1.1 研究背景及意义

计算机视觉[1][2][3]在图像识别、目标检测、场景理解等方面取得了重要突破，对人工智能技术在各领域的应用产生了深远的影响。如今，人们不再满足于物体检测、识别这类简单的视觉理解任务，而是追求更高层次的视觉场景推理任务。例如，对于给定的一张图片，人们不仅希望准确地检测识别图片中的各对象，还希望可以理解对象之间的关系（视觉关系检测），并且生成文字描述（看图说话/图像说明）。基于此，研究者们提出了场景图（Scene Graph）的概念：一种结构化、语义化的场景表示方式，用于清晰地表达一个场景中的所有物体、物体的属性，以及物体间的关系[4]。场景图的强大能力一方面体现在它可以将 2D/3D 图像[5]和视频编码为抽象的语义信息，另一方面，它并不会限制对象的类型和属性，也不限制对象之间的关系[6]。场景图生成（Scene Graph Generation, SGG）是指将图像自动映射为语义结构场景图的任务，要求对检测到的对象和它们之间的关系进行正确标注。

作为一种强大的语义表示方法，场景图已经引起了计算机视觉领域广泛的关注。早在 2015 年，利用图像中不同对象之间的视觉特征和它们之间的关系来实现多种视觉任务的想法已被提出。人们希望通过挖掘场景中对象的关系来完成一系列视觉任务，包括动作识别、生成图像字幕、机器人导航等相关计算机视觉任务。这种方法已被证明可以显著提高相关视觉任务的性能，并且有效增强人们理解和推理视觉场景的能力。随后，Johnson 等人将视觉关系纳入场景图理论中，并正式提供了场景图的定义[7]。人们可以通过从现实世界场景图的数据集中手动生成场景图，从而捕捉复杂场景的详细语义信息。从那时起，场景图的研究受到了广泛关注。这些场景理解的研究大大推动了计算机视觉、自然语言处理以及它们的跨领域理解等各种任务的发展。

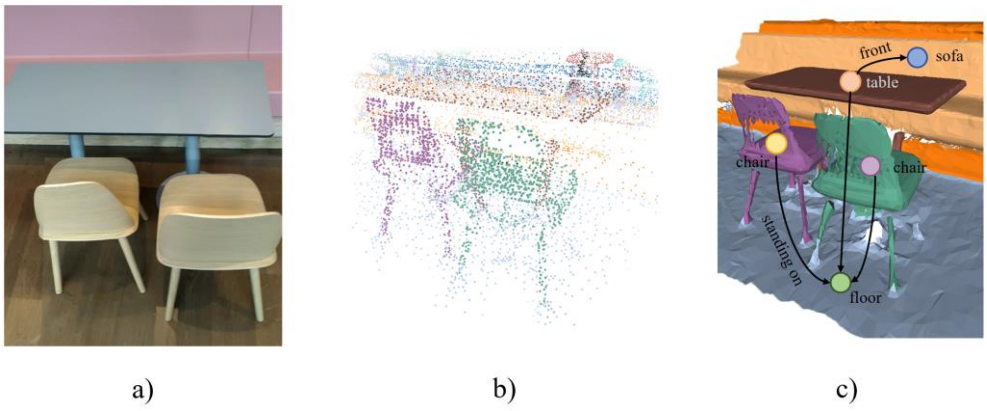

图 1 一个简单的场景图生成流程，其中 a) 是一个真实室内场景的分割，通过 RGB-D 相机扫描并重建得到 b) 3D 点云，点云通过场景图生成算法后得到 c) 重建的三维场景和语义场景图。

与从图像生成 2D 场景图类似，3D 场景图生成算法目标是从 3D 场景中构建提供准确量化 3D 场景中对象关系的表示方式，具有重要的理论价值和实际意义。一方面，场景图表示方法可以为计算机视觉领域的物体识别、场景理解和智能交互等应用提供更加丰富的语义信息。另一方面，场景图生成算法相比传统的识别算法，可以更好地理解 3D 场景中物体之间复杂的三维视觉关系。显然，与 2D 场景理解相比，3D 场景理解更具挑战性。3D 场景下的理解任务不仅需要处理复杂多样的 3D 场景数据，还需要学习多出一个维度带来的更复杂的

物体与关系的特征嵌入。然而，3D 场景理解为实例分割[8][9]、语义分割[10][11]以及 3D 物体检测和分类[12][13]等任务提供了大量的便利，在许多前沿领域具有广泛的应用前景，如虚拟现实、增强现实、无人机导航等。因此，3D 语义场景图生成算法具有重大研究意义。

1.2 国内外研究现状

受图像检索的启发，Johnson 等人[7]最早提出了场景图（Scene Graph）的概念。场景图是一种结构化的场景表示方法，作为一种语义描述图像的表示形式。场景图的基本构成包括节点和边：节点代表场景中的物体，边代表物体间的关系。早期的场景图研究主要集中在 2D 图像上，研究者们通过构建 2D 场景图来实现对 2D 图像的结构化分析与理解。Visual Genome[14]是一个大规模的数据集，用于在图像上注释场景图。这项工作催生了一系列基于深度学习的场景图预测技术，如 Xu 等人[15]的基于消息传播机制的场景图生成算法、Qi 等人[16]的注意力关系网络和 Yang 等人[17]的 Graph R-CNN 算法。这些方法提出了多种高效的图传播策略和场景预测方法，例如消息传递、图卷积网络和注意力机制等，为 2D 场景图生成和计算机视觉领域提供了许多前沿的启发和贡献。

直到最近，3D 场景图的研究工作才开始出现并且逐渐增多。目前主要有两大类 3D 场景图生成工作。第一类基于图像进行研究。Armeni 等人[18]第一次利用 3D 网格和 RGB 全景图像来构建 3D 场景图，完成了开创性的工作。Kim 等人[19]提出了基于 SLAM 的局部图融合全局图的框架，用于生成 3D 场景图。第二类方法直接对 3D 数据进行处理。其中，最具代表性的是 Wald 等人[23]的场景图预测网络（SGPN），它以扫描的 3D 点云作为输入，使用图卷积网络对场景图建模和预测。此外，Wald 等人还提出了 3DSSG 数据集[23]，为 3D 场景图研究领域奠定了基础。此后，大量工作都是在 3DSSG 数据集上进行的，如 Zhang 等人[20]利用先验知识融入图形表示中来优化 SGPN 的框架。Zhang 等人[21]进一步探索节点和边的演化过程，并使用两个注意力掩码取得了一定的提升。Wu 等人[22]将多种增量模块融入 SGPN 的框架，提出了场景图融合网络（SGFN）并在 3DSSG 数据集下实现了优越的效果。

1.3 本论文的主要工作

针对复杂而多样的 3D 场景，本文在 SGPN 框架的基础上，提出了基于注意力的场景图预测网络（Attention-based Scene Graph Prediction Network, ASGPN），以提升关键视觉线索的捕捉能力。

本工作的主要贡献如下：1）针对物体特征，本文采用了一种逐特征的注意力机制来捕捉物体和关系之间的权重信息，使网络在特征交互中更加关注关键的特征信息；2）针对关系特征，本文引入了一种利用节点特征的注意力掩码，以此协助关系特征在多层感知机中更精确地学习。相比与 SGPN 原本的图卷积网络部分，本文优化的注意力图卷积部分在特征交互上有明显的进步。

本工作在包含了 1482 个室内场景的 3DSSG 数据集中进行了模型的验证和对比。对于场景图预测物体和关系的能力评估，本文采用常见的 Top-K 召回率指标。ASGPN 相比原始 SGPN 平均提升了 19.68%，相比 EdgeGCN 平均提升了 10.75%。同时，本文对 ASGPN 提出的优化模块分别进行了消融实验，以此证明各贡献点的有效性。最后，本文对生成的场景图进行了可视化展示，包括原始的相机扫描、分割场景的点云以及算法生成的语义场景图。

1.4 论文的组织架构

本文的组织架构如下：第一章简要介绍了场景图在计算机视觉和深度学习领域的背景知识，以及场景图的实际应用前景。第二章详细分析了 3D 场景图生成的相关工作，并指出了其在 3D 场景理解中的不足。第三章提出了一种基于注意力机制的 3D 场景图生成算法，名为基于注意力的场景图预测网络（ASGPN），并对其模型结构进行逐步分析。第四章介绍了本工作使用的数据集，探讨了算法的系统实现、实验结果、基线方法对比以及可视化，并进行消融实验以证明贡献点的有效性。最后在总结部分讨论了本工作的优点和局限性，并提出了未来研究的方向。

2 相关工作

2.1 3D 点云处理技术

3D 点云是一种表示三维空间中物体表面的数据结构，通过采集大量具有三维坐标的点来描述物体的形状。3D 点云数据可以从各种传感器（如激光雷达、RGB-D 相机）和方法（如立体视觉、多视角图像）中获得。在计算机视觉和机器人领域，3D 点云处理技术在目标识别、场景重建、自动驾驶等方面具有广泛的应用[24]。

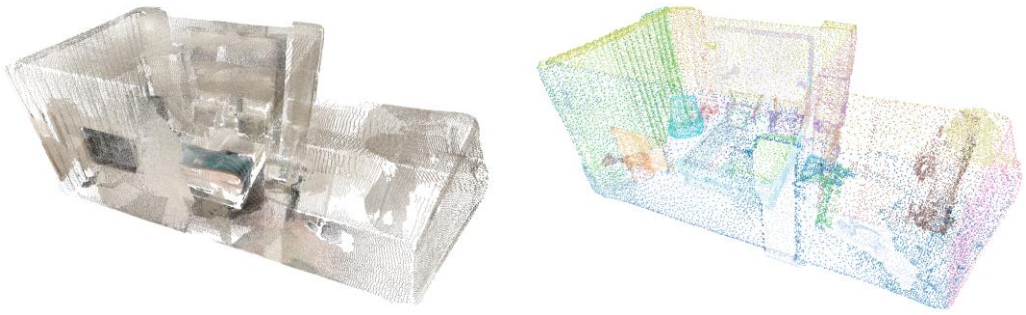

图 2 室内 3D 点云示例。左图是来自雷达或 RGB-D 相机扫描后（重建后）的原始 RGB 点云。右图是已标注的重建点云。

早期的 3D 点云处理方法主要基于传统计算机视觉技术如滤波、降采样、法向量估计、ICP 算法等。这些方法在特定应用场景中取得了一定的成果，但在处理大规模、噪声严重、具有复杂结构的点云数据时面临诸多挑战。随着深度学习技术发展，人们利用具有更强的表征能力和鲁棒性的深度神经网络来处理更复杂的 3D 点云数据。Qi 等人[10]设计的 PointNet 是第一个直接处理原始点云数据的深度学习网络。它通过对输入点云进行全局特征提取，实现了端到端的点云分类和分割。PointNet 的出现开创了基于深度学习的 3D 点云处理方法的研究。随后的 PointNet++[11]在 PointNet 的基础上引入了分层采样和局部特征融合的策略，进一步提升了分类的精度。Wang 等人[25]的 DGCNN 通过构建 k 近邻图和学习局部特征，设计了一种基于动态图卷积的 3D 点云处理网络，在三维物体分类任务上取得了优异的性能表现。

2.2 图神经网络

图神经网络（GNN）是一类用于处理图结构数据的神经网络模型[26]。由于现实世界中许

多复杂系统可以表示为图结构，如社交网络、交通网络和知识图谱等，因此 GNN 在多个领域具有广泛的应用价值。GNN 将图定义为一种由节点（node）和边（edge）组成的网状数据结构。节点和边代表的实际意义会被编码成特征嵌入。GNN 的核心思想是通过信息传播和特征聚合来学习节点和边的表示，从而捕捉节点之间的关系和拓扑结构。

随着深度学习技术的发展，近年来越来越多基于图神经网络的优化模型提出。Kipf 等人[27]首先提出了将卷积架构用于图神经网络上，提出了图卷积网络。它通过定义一种基于邻居节点特征的卷积操作，实现了局部信息的聚合，并在多个图相关任务上取得了显著的性能提升。Hamilton 等人[28]提出了 GraphSAGE，一种采用采样和聚合策略、能够在大规模图数据上进行高效训练的图神经网络。GraphSAGE 为 GNN 在大规模图处理任务中的应用奠定了基础。Veličković 等人[29]将领域内最新的注意力机制与 GNN 进行融合，提出了图注意力网络（Graph Attention Networks, GAT）。通过为邻居节点分配不同的权重，GAT 可以捕捉图中的有向边和异构结构，并且在多个图相关任务中表现出了优越的性能。Xu 等人[30]提出了图同构网络（Graph Isomorphism Network, GIN），通过引入一种特殊的聚合函数和归一化策略，GIN 能够捕捉图中的拓扑信息和节点特征。这些基于图结构的神经网络在许多分类任务上取得了优秀的性能，适用于各种复杂的数据结构，在实际应用场景中发挥着巨大作用。

然而这些经典的图神经网络在适应学习复杂的 3D 场景表示表现出的性能不佳。由于 GNNs 主要关注局部结构，它们可能难以捕捉 3D 场景中的全局和长距离视觉线索。

2.3 注意力机制

注意力机制是一种模拟人类注意力分配的计算模型，在深度学习领域取得了广泛应用。通过为输入数据分配不同的权重，注意力机制能够关注到重要的信息并忽略不相关的信息。注意力机制最早由 Bahdanau 等人[31]在深度学习领域使用，起初用于自然语言处理相关任务（如机器翻译）。在这些任务中，注意力机制可以根据源语言句子中的每个单词与目标语言单词之间的相关性为每个单词分配不同的权重。这种权重分配策略有助于捕捉长距离依赖关系和处理不同长度的输入序列。真正使注意力机制在深度学习领域家喻户晓的是 Vaswani 等人[32]提出的 Transformer 模型，一种利用多头注意力机制的深度网络。多头注意力机制通过多个注意力头，网络可以从输入数据中捕捉到更丰富的信息。每个注意力头都可以关注不同的输入特征，从而使模型能够同时考虑多种依赖关系和相关性。Transformer 在多个领域取得了显著的性能提升，包括机器翻译等自然语言处理任务和图像分类、目标检测和语义分割等视觉任务。Veličković 等人[29]随后将注意力机制带到了图领域，提出的图注意力网络（GAT）在图结构任务中表现出了优越的性能。此外，注意力机制还被用于卷积神经网络（CNN）中，通过关注不同区域的特征来提高模型的表达能力[33]。在场景图任务中，Wu 等人[22]利用图注意力机制优化了特征交互能力，Zhang 等人[21]利用节点和边的孪生关系进一步探索特征嵌入的演化过程。

3 基于注意力的场景图预测网络

3.1 问题描述

本节首先给出场景图生成的目标和数学表示方法。在 3D 场景图生成任务中，一个原始的场景 S 会被切割成若干个类别无关的场景分割集合，即 $S = \{s_1, \dots, s_n\}$。每个场景分割 s_i 包含一组 3D 点集 $\mathcal{P}$（又称点云），注意力场景图预测网络（Attention-based Scene Graph

Prediction Network, ASGPN）的目标是生成一个图结构 $\mathcal{G}=\{\mathcal{N},\mathcal{R}\}$，其中场景里的物体表示为 $\mathcal{N}$，物体间的关系表示为 $\mathcal{R}$。

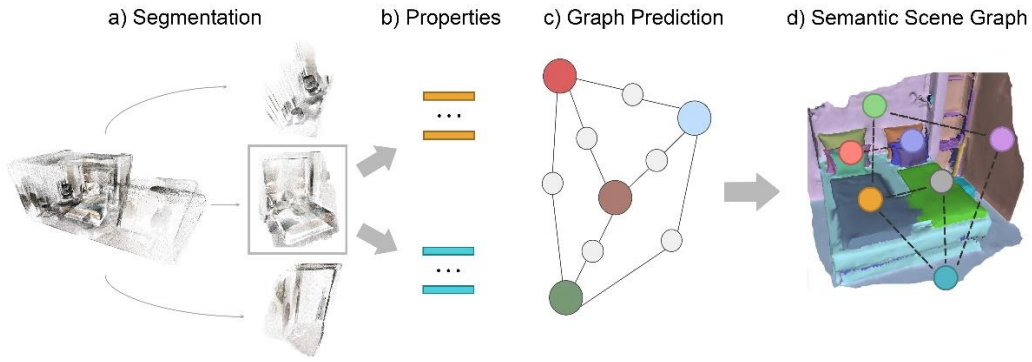

图 3 ASGPN 工作流程示例。原始的场景扫描首先被切割成不同的 a) 场景分割进行训练和预测。分割点云会被提取出初始的 b) 节点特征和边特征，这些特征进入 c) 基于注意力的图卷积部分进行学习和预测，最终得到 d) 重建的已标注的场景和语义化的场景关系图。

本文定义物体（object）是“椅子”、“沙发”等常见室内实体；边（或称谓语，predicate）是“悬挂在”、“支撑着”等物体间的物理关系。一个关系（relationship）指一个完整的主语、谓语和宾语对应的三元组，如关系“<椅子，在左边，桌子>”表示语义信息“椅子在桌子左边”。ASGPN 旨在根据场景点云正确地分类场景中物体、谓语和关系，并构建出对应的 3D 场景图 $\mathcal{G}$。

3.2 模型结构

本文提出了一种基于注意力的图卷积网络，该网络使用两种不同的注意力机制，通过融合节点特征和边特征来学习更为丰富的图表示。这种新颖的注意力图卷积网络将会用在基于注意力的场景图预测网络（ASGPN）上，替换 SGPN[23]框架下原有的三元组图卷积网络。ASGPN 首先处理点云数据表示的 3D 场景得到原始特征嵌入，然后经过一个注意力图卷积网络（见图 3.c 和图 4）进行特征交互，最后对节点和边进行分类和预测。以下是模型结构的详细介绍。

3.2.1 点云特征提取

模型得到的最初始输入是场景分割 s_i 的 3D 点云 $\mathcal{P}$。和 2.1 节中介绍的一样，点云是三维物体的采样表示，是由若干无序的点组成集合。一般来说，每个点至少包含位置信息，即 XYZ 的坐标。ASGPN 的第一个任务是将这些 3D 点云抽象成特征嵌入（见图 3.b）。常见的办法是采用 PointNet 对点云进行特征提取。给定一个场景 s_i，首先要对场景中的每个实例（物体）i 分别提取点集。这一步 ASGPN 和大部分 3D 场景图模型[21][22][23]一样，通过设计掩码矩阵 $\mathcal{M}$ 仅保留与实例 i 相关的点。具体来说，对于每个实例 i，其对应的点集 $\mathcal{P}_i$ 由以下公式计算得出：

$$\mathcal{P}_i=\left\{\delta_{m_k i}\odot p_k\right\}_{k=1,|\mathcal{P}|}, \tag{1}$$

其中，p 和 m 分别是点云 $\mathcal{P}$ 和掩码矩阵 $\mathcal{M}$ 的实例，$\odot$ 表示逐元素乘法，$|\cdot|$ 表示 $\mathcal{P}$ 包含点

的数量。δ 表示克罗内克（Kronecker）函数，它存在以下关系：

$$\delta_{ij} = 1 \Leftrightarrow i = j. \tag{2}$$

在这里，它用于检查点 p_k 是否属于实例 i，满足时 $\delta_{m_k i} = 1$，否则 $\delta_{m_k i} = 0$。通过与点 p_k 逐元素相乘，模型可以构建与实例 i 相对应的点集 $\mathcal{P}_i$。对于节点特征提取，本文使用第一个 PointNet 将每一个实例对应的点集 $\mathcal{P}_i$ 编码为表示物体的原始形状的潜在特征：

$$v_i = f_v(\mathcal{P}_i), \tag{3}$$

其中 $f(\cdot)$ 代表一个简化的 PointNet。这个简化的 PointNet 去除了两次空间变换网络（Spatial Transformer Networks）的操作，而这是基于模型的精度和训练代价综合考虑的（在生成边关系特征时会进一步阐释）。具体来说，这个简化版 PointNet 包含了三个一维卷积层。对于每个卷积层，模型都是用大小为 1 的卷积核，并且采用整流线性单位函数 ReLU 来激活：

$$ReLU(x) = max(0, x). \tag{4}$$

点集特征经过最后一层卷积层后，执行最大池化操作以保留最大特征值，并将结果调整为所需的输出形状，最终生成具有全局特征的输出向量。PointNet 能够在无序的点云数据上直接学习到有意义的特征，从而应对各种三维数据处理任务。

对于边特征的生成，本文通过计算节点 i 和 j 各自的 3D 包围盒（Bounding Box）$\mathcal{B}_i$ 和 $\mathcal{B}_j$ 的并集，从点云 $\mathcal{P}$ 中提取属于这两个包围盒的点集 $\mathcal{P}_{ij}$：

$$\mathcal{P}_{ij} = \{p_k \,|p_k \in (\mathcal{B}_i \cup \mathcal{B}_j)\}_{k=1,|\mathcal{P}|}, \tag{5}$$

将点集 $\mathcal{P}_{ij}$ 与相应的掩码矩阵 $\mathcal{M}_{ij}$ 拼接作为第二个 PointNet 的输入：

$$e_{ij} = f_e([\mathcal{P}_{ij}, \mathcal{M}_{ij}]), \tag{6}$$

该 PointNet 依旧延续了上述的设定。其中，掩码矩阵 $\mathcal{M}_{ij}$ 的值根据点 p_k 对应的对象是 i 还是 j 来确定：如果 p_k 对应对象 i，则 $\mathcal{M}_{ij}$ 的值为 1；如果 p_k 对应对象 j，则 $\mathcal{M}_{ij}$ 的值为 2；其他情况下，$\mathcal{M}_{ij}$ 的值为 0。ASGPN 在处理关系特征时，保留边上下文 $\mathcal{P}_{ij}$ 的方向信息对于推断诸如“左侧”或“右侧”等位置关系非常重要。因此，简化版的 PointNet 去除了空间变换的操作。

经过了上述两个 PointNet 的特征提取，初始的点云 $\mathcal{P}$ 被编码成了节点特征 $v_i \in \mathbb{R}^{O \times d_n}$ 和边特征 $e_{ij} \in \mathbb{R}^{T \times d_e}$ 交付给图卷积部分作进一步的特征学习，其中 O 和 T 分别是场景分割 s_i 里物体/实例的数量和谓语/边的数量。

3.2.2 节点逐特征注意力

得到物体和物体的关系的抽象初始特征后，我们构建 ASGPN 的注意力图卷积部分（见图 3.c）。这一部分的网络会将节点和边的特征不断演化，使其适用于下游分类任务。

为了充分利用边的特征信息，ASGPN 引入了逐特征的注意力机制（Feature-wise Attention）[22]。在 (3) 式中得到的节点特征 v_i 会根据一个稀疏的邻接矩阵索引为源节点 $x_i \in \mathbb{R}^{T \times d_n}$ 和目标节点 $x_j \in \mathbb{R}^{T \times d_n}$。逐特征注意力模块（见图 5 左半部分）接收源节点特征 x_i、(6) 式中得到的边特征 e_{ij} 和目标节点特征 x_j 作为输入，这代表了一个 <主语，谓语，宾语> 的三元组语义结构。

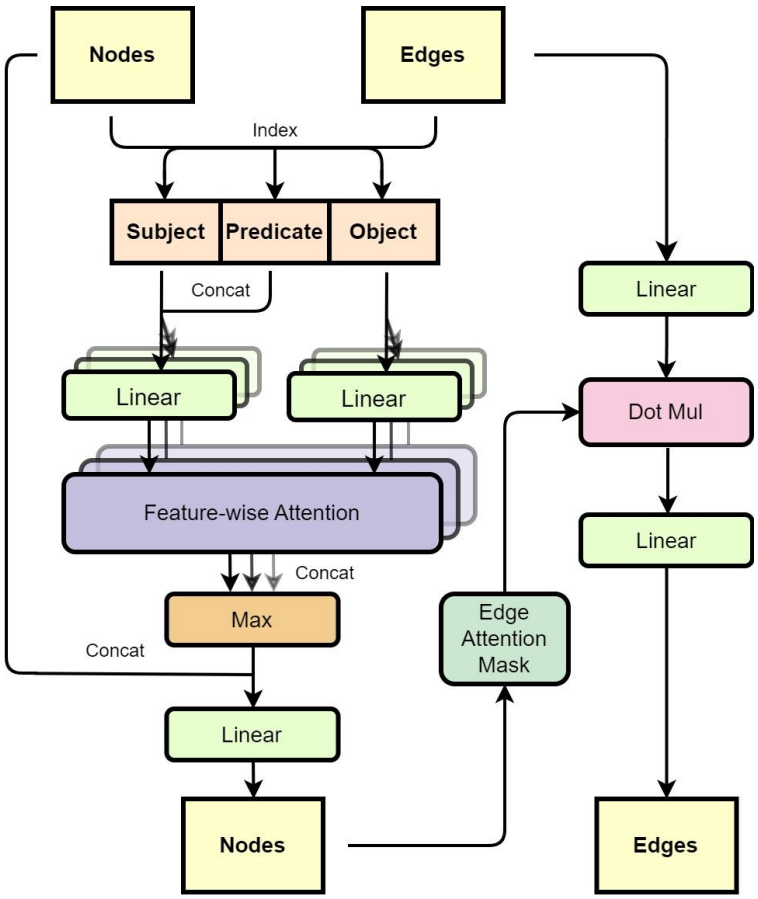

图 4 基于注意力的图卷积部分网络结构。该模块接收节点特征矩阵（Nodes）和边特征矩阵（Edges）作为输入，并输出形状相同的更新后的节点特征矩阵和边特征矩阵。图中左半部分对应 3.2.2 节中的节点逐特征注意力，右半部分对应了 3.2.3 节中的边注意力掩码。

为了使特征的演化适用于注意力机制，本文首先对源节点特征 x_i、边特征 e_{ij} 和目标节点特征 x_j 进行了线性变换，得到查询向量 q_i、边向量 $\widetilde{e_{ij}}$ 和值向量 v_j：

$$q_i = {W_Q}^T x_i,$$
$$\widetilde{e_{ij}} = {W_E}^T e_{ij},$$
$$v_j = {W_V}^T x_j, \tag{7}$$

其中 $W_Q \in \mathbb{R}^{d_n \times d_{q/2}}$、$W_E \in \mathbb{R}^{d_e \times d_{q/2}}$ 和 $W_V \in \mathbb{R}^{d_n \times d_v}$ 分别为源节点特征 x_i、边特征 e_{ij} 和目标节点特征 x_j 的投影矩阵。接着，模型将查询向量 q_i、边向量 $\widetilde{e_{ij}}$ 进行连接，并通过一个多层感知机（MLP）计算注意力分数 α_{ij}：

$$\alpha_{ij} = g_\alpha([q_i, \widetilde{e_{ij}}]), \tag{8}$$

其中 $g(\cdot)$ 代表使用整流线性单位函数 ReLU 激活的 MLP，下同。和常见的注意力[32]的做法类似，本文使用 softmax 函数计算归一化的注意力权重 β_{ij}：

$$\beta_{ij} = \frac{\exp(\alpha_{ij})}{\sum_{k \in \mathcal{N}(i)} \exp(\alpha_{ik})}, \tag{9}$$

其中 $\mathcal{N}(i)$ 表示节点 i 的邻居节点集合。这里得到的注意力权重 β_{ij} 将和值向量 v_j 做逐元素

乘法，由此得到逐特征的注意力值：

$$\text{Atten}(x_i,\, e_{ij},\, x_j) = \beta_{ij} \odot v_j. \tag{10}$$

受 Vaswani 等人[32]的启发，本文将注意力扩展成多头（Multi-head）的形式，以便从输入数据中捕捉到更丰富的信息。注意力的每个头可以学习到不同的特征表示，这有助于模型学习更复杂的输入数据模式，同时可以提高模型的梯度稳定性。模型将源节点特征 x_i、边特征 e_{ij} 和目标节点特征 x_j 在其各自的维度上分成 h 个头部，每个头部将应用 (10) 式的形式分别计算注意力值。最后，这些头部的注意力值将会重新拼接回维度 d_t，从而获得多头逐特征注意力值：

$$\text{M-Atten}(x_i,\, e_{ij},\, x_j) = \left[\text{Atten}(x_{im},\, e_{ijm},\, x_{jm})\right]_{m=1}^{h}, \tag{11}$$

其中 $m = 1, \ldots, h$。对于第 l 层节点特征更新，模型按照 (11) 式依次计算其所有邻居的多头注意力值。这个多头注意力值的集合会取一个最大聚合值，然后和节点 i 的原始特征进行直接拼接，起到类似残差连接的效果。最后，ASGPN 通过一个 MLP 计算更新后的第 $l+1$ 层的节点特征：

$$\begin{aligned} v_i^{l+1} &= g_x\left(\left[v_i^l, \max_{j\in\mathcal{N}(i)}\left(\text{M-Atten}(x_i^l,\, e_{ij}^l,\, x_j^l)\right)\right]\right) \\ &= g_x\left(\left[v_i^l, \max_{j\in\mathcal{N}(i)}\left(\left[\frac{\exp\left(g_\alpha\left(\left[{W_Q}^T x_{im}^l, {W_E}^T e_{ijm}^l\right]\right)\right)}{\sum_{k\in\mathcal{N}(i)}\exp\left(g_\alpha\left(\left[{W_Q}^T x_{im}^l, {W_E}^T e_{ikm}^l\right]\right)\right)} \odot {W_V}^T x_{jm}^l\right]_{m=1}^{h}\right)\right]\right). \end{aligned} \tag{12}$$

注意，对于给定场景分割 s_i，其更新前和更新后的节点特征向量始终保持形状一致，即 $v_i^{l_k} \in \mathbb{R}^{O\times d_n}$。边特征的更新也遵循同样的规律，即 $e_{ij}^{l_k} \in \mathbb{R}^{T\times d_e}$。

3.2.3 边注意力掩码

ASGPN 利用节点特征映射来捕捉源节点和目标节点之间的相互作用，以便在它们连接的边上建模。这种注意力机制通过一个多维注意力掩码 $\mathcal{A}_x \in \mathbb{R}^{T\times d_{e/2}}$ 实现（见图 4 右半部分），其中 T 是一个场景里边的数量。具体来说，给定节点 i 和节点 j 以及连接它们的有向边 $E(i,j)$，可以通过含有 ReLU 激活的单层感知机学习源节点和目标节点的拼接的演化特征 $\hat{x}_{ij}$：

$$\begin{aligned} \hat{x}_{ij} &= ReLU\left(\left[{W_N}^T x_i, {W_N}^T x_j\right]\right) \\ &= max\left(0, \left[{W_N}^T x_i, {W_N}^T x_j\right]\right), \end{aligned} \tag{13}$$

其中 $W_N \in \mathbb{R}^{d_n\times d_{n/2}}$ 是节点特征源节点 i 和目标节点 j 共享的投影矩阵。这个演化特征可以用来学习节点之间的边方向节点驱动交互程度。本文采用含有 sigmoid 激活函数的 MLP 来计算该交互程度，并将计算得到的数值称为边注意力掩码：

$$\begin{aligned} \mathcal{A}_x &= \sigma\left({W_M}^T \hat{x}_{ij}\right) \\ &= \frac{1}{1+\exp(-{W_M}^T \hat{x}_{ij})}, \end{aligned} \tag{14}$$

其中 $W_M \in \mathbb{R}^{d_e\times d_{e/2}}$ 是拼接演化特征 $\hat{x}_{ij}$ 的投影矩阵，$\sigma(\cdot)$ 是 sigmoid 激活函数。对于第 l 层

边特征的更新，本文使用两个 MLP 对边特征进行处理，以得到更高级的边表示。初始的边特征 e_{ij} 会先通过第一个 MLP 演化，然后与上述边注意力掩码进行逐元素相乘，再通过第二个 MLP 完成最终的更新：

$$\begin{aligned} e_{ij}^{l+1} &= g_{e2}\left(g_{e1}\left(e_{ij}^{l}\right) \odot \mathcal{A}_x\right) \\ &= g_{e2}\left(g_{e1}\left(e_{ij}^{l}\right) \odot \frac{1}{1+\exp\left(-W_M{}^T max\left(0,\left[W_N{}^T x_i, W_N{}^T x_j\right]\right)\right)}\right). \end{aligned} \tag{15}$$

由此，ASGPN 的注意力图卷积部分完成了第 l 层的节点特征和边特征向下一层的更新。

3.2.4 类别预测和损失计算

由注意力图卷积部分更新的节点特征和边特征需要从特征空间中映射回类别。与常见的 3D 场景图生成模型[22][23]类似，本文在 ASPGN 的尾端采用两个 MLP 分类器对节点特征和边特征进行分类。两个分类器会分别计算物体的损失值 $\mathcal{L}_{obj}$ 和谓语的损失值 $\mathcal{L}_{pred}$。这里模型采用负对数似然损失（Negative Log Likelihood Loss）作为计算：

$$\text{nll\_loss}(x, y) = -\frac{1}{N}\sum_{i=1}^{N} \log(x_i, y_i), \tag{16}$$

其中，N 表示样本数，x_i, y_i 表示第 i 个样本在真实标签 y_i 上的预测概率值。与一些繁杂的模型融合策略[34]不同，ASPGN 不会对某一个模块进行预训练。我们不考虑某个局部的具体拟合状态，因此任何有利于最终损失下降的迭代都会被保留。整个模型会高效地在 3DSSG 数据集上根据物体和谓语的加权损失 $\mathcal{L}$ 进行端到端学习：

$$\mathcal{L} = \alpha\mathcal{L}_{obj} + \beta\mathcal{L}_{pred}, \tag{17}$$

其中 α 和 β 是超参数。

4 实验评估和可视化

在这一章中，本文将评估提出的基于注意力的场景图预测网络（ASGPN）。这一部分的实验将会在 Wald 等人[23]提出的 3DSSG 数据集上进行训练和验证。4.1 节中将给出本文进行程序运行的物理实验环境和虚拟实验环境。4.2 节对实验所使用的 3DSSG 数据集进行全面的介绍。4.3 节将介绍用于对比 ASGPN 的各种基线方法。4.4 节展示了如何将 3DSSG 数据集做预处理并应用于深度网络学习。4.5 节将介绍评估场景图预测的指标计算方法。4.6 节给出了详细的实验细节、实验结果以及生成的场景图可视化演示。

4.1 实验环境

本次实验全程运行在 Ubuntu 18.04 系统的服务器上。该服务器配备了一张 Nvidia RTX2080Ti 显卡（12GB 显存），CPU 为 Intel i9-9900K 3.60GHz，内存为 64GB，硬盘容量为 1TB。我使用 Anaconda 管理的 Python 3.8.8 编写所有的程序代码，并使用运行 CUDA 加速的 PyTorch 1.8 框架进行深度学习网络的编写与测试。运行本项目的具体虚拟环境所需的全部第三方库如表 1 所示。

表 1　实验环境所需 Python 第三方库

库名	版本
PyTorch	1.8.0
CUDA	10.2.89
cuDNN	7.6.5
NumPy	1.23.5
ONNX Runtime	1.12.1
TensorBoard	1.15.0
Trimesh	3.9.13
Open3D	0.16.0
Plyfile	0.7.4
PyTorch Cluster	1.5.9
PyTorch Geometric	1.7.0
PyTorch Scatter	2.0.6
PyTorch Sparse	0.6.9
tqdm	4.60.0

其中，PyTorch 是目前最广泛使用的深度学习框架，提供了用于构建和训练神经网络的工具和 API。NVIDIA 的 CUDA 和 cuDNN 加速了基于 CUDA 平台的深度学习计算，使 PyTorch 利用 GPU 上的并行计算能力成为可能。NumPy 是一个用于科学计算的 Python 库，提供了用于处理数组和矩阵的功能。ONNX Runtime 是一个用于运行 Open Neural Network Exchange 模型的跨平台推理引擎。Trimesh 是一个用于处理 3D 几何的 Python 库，支持点云、网格和其他 3D 数据结构的处理，本文使用 Trimesh 来读入点云数据。Open3D 也是一个用于处理 3D 数据的开源库，提供了用于点云、三角网格和其他 3D 数据结构的处理方法。Plyfile 是一个用于读写 PLY 格式 3D 模型文件的 Python 库。PyTorch Cluster、PyTorch Geometric、PyTorch Scatter 和 PyTorch Sparse 都是 PyTorch 框架下的拓展 API，提供了处理图和点云数据、张量散射和聚合操作、处理稀疏张量和稀疏矩阵乘法等功能。

本工作按照 Wu 等人在 Github 仓库[35]中的指引在服务器上配置了上述环境。

4.2　3DSSG 数据集

3DSSG 数据集是由 Wald 等人[23]提出的 3D 语义场景图数据集，在 3D 场景图领域中被认为是最具代表性且应用最为广泛的数据集。3DSSG 数据集提出后，3D 场景图生成相关研究[22][20][21]逐渐增多，并且大多都在该数据集上进行评估，逐渐形成了统一的度量标准。3DSSG 数据集基于真实世界的室内环境，因此更具代表性和可靠性。这也使得使用该数据集的研究成果更易于推广应用于实际场景。

3DSSG 数据集基于 3RScan[36]，一个大规模真实世界数据集，包括 478 个自然变化的室内环境的 1482 个 3D 重建。3DSSG 为研究者提供了丰富的 3D 语义场景图，有助于更好地理解和分析复杂的三维空间。数据集中的场景图由节点和边组成，节点表示 3D 扫描中的特

定 3D 对象实例，而边则定义了节点之间的语义关系。3DSSG 数据集包括 1482 个场景图，其中有 4.8 万个对象节点和 54.4 万条边。表 2 中展示了当前主流的两个 3D 场景图数据集对比，展示了 3DSSG 的优越性和应用前景。

表 2　3D 语义场景图数据集对比

数据集	规模	实例	物体类别	关系类别
Armeni et al. [5]	35 buildings 727 rooms	3k	28	4
3DSSG [23]	1482 scans 478 scenes	48k	534	40

场景图中的节点代表 3D 对象实例，每个实例都分配给一个 3D 场景。属性是描述对象实例的语义标签，包括静态和动态属性。由于对象实例数量庞大，属性的语义多样性很高，因此有效的提取和注释设计至关重要。属性不仅描述了对象实例的视觉和物理外观，还包括了对象之间的关系和相互作用。除了属性，3DSSG 数据集还提供丰富的关系类型，主要可分为三类：空间关系、支持关系和比较关系。这些关系有助于更好地理解场景中对象之间的相互作用和联系。

图 5 3DSSG 数据集室内场景扫描（点云）示例

3DSSG 数据集的原始格式是 PLY 格式的点云文件（如图 5 所示）。和 Wu 等人[22]的工作一样，本文使用了包含 160 个物体类别和 26 和谓词类别的语义标注的 3DSSG 数据集来评估本次实验。表 3 和表 4 分别展示了实验中 3DSSG 数据集所包含的物体类别（部分）和关系类别（部分）。

表 3　3DSSG 物体类别（部分）

类别 1-5	类别 6-10	类别 11-15	类别 16-20
armchair	counter	laptop	shoe rack
backpack	cup	laundry basket	shoes
bag	cupboard	light	showcase
ball	curtain	machine	shower
bar	cushion	magazine rack	shower curtain

表 4　3DSSG 谓词类别（部分）

类别 1-5	类别 6-10	类别 11-15	类别 16-20
supported by	close by	lower than	lying on
left	inside	same symmetry as	hanging on
right	bigger than	same as	connected to
front	smaller than	attached to	leaning against
behind	higher than	standing on	part of

4.3　基线方法

为了评估本文提出的 ASGPN 模型，本文将使用 6 个不同的图神经网络模型和领域内最前沿的基于深度学习的 3D 场景图生成模型与 ASGPN 进行对比。为了保证实验的公平，本文将统一使用 3DSSG 数据集进行训练和验证，并使用相同的指标作比较。因此，所有的模型均会调整为基于点云的 3D 场景图生成框架，如图 4 所示。初始的 3D 点云会经过分割并通过两个 PointNet 提取初始特征，然后进入基于图的深度网络中进行学习，最后使用两个 MLP 进行分类和预测。即，本工作只更换图神经网络部分（图 4.c）进行对比。下面是每个基线模型的详细介绍。

1) **不使用图神经网络（w/o GNN）。**模型直接学习两个 PointNet 提取后的初始特征。这个实验作为空白参照。

2) **图卷积网络[27]（GCN）。**GCN 将卷积的概念引入图神经网络的每层特征更新之中。通过利用了图结构中的局部邻域信息来学习和更新节点（物体）和边（关系）的表示，GCN 可以捕获物体之间的关系以及关系之间的依赖性，从而在场景图上进行更精确的推理。

3) **图采样与聚合[28]（GraphSAGE）。**GraphSAGE 使用可学习的聚合函数（如平均、最大池化或长连接）来聚合邻居节点的特征。与 GCN 不同，GraphSAGE 不依赖于邻接矩阵的特征，而是使用归纳学习方法来生成节点嵌入，因此能够更好地捕获物体特征和关系特征的局部邻域信息，从而使得场景图中的节点和边在特征空间中进行更新。

4) **图注意力网络[29]（GAT）。**GAT 可以捕获物体与物体之间以及物体与关系之间的复杂依赖关系。场景图生成通常涉及对图形结构数据的处理，其中物体和关系作为节点，它们之间的相互作用作为边。GAT 通过利用注意力机制可以对这些结构化数据进行高效建模。

5) **场景图预测网络[23]（SGPN）。**SGPN 是领域内被广泛采用的 3D 场景图生成模型。该模型使用两个 PointNet 提取特征，使用一个三元组图卷积网络进行特征交互，最后使用两个 MLP 进行节点和关系分类。其中的三元组 GCN 将普通 GCN 中独立更新的节点特征和边特征索引到了一个共同更新的 <主语，谓语，宾语> 三元组中，再使用 MLP 进行迭代，从而更深入地学习了场景的拓扑信息。同时，网络还引入了残差连接，进一步增强了模型的鲁棒性。

6) **基于边的图卷积神经网络[21]（EdgeGCN）。**EdgeGCN 是对 SGPN 框架的优化，深

入研究了图网络中节点和边的孪生关系，并在图卷积网络中加入了互相协助演化的注意力掩码。这种网络可以充分利用多维边特征来显式地建模场景中物体之间的关系，有助于捕捉 3D 场景中的复杂结构。

4.4 数据预处理

3DSSG 数据集基于 3RScan[36]，3DSSG 是一个庞大而复杂的数据集，包括 478 个自然变化的室内环境的 1482 个 3D 重建。原始的数据集大小约为 94GB，其各文件组织形式如表 5 所示。从 Wald 等人提供的项目[23]中可以得到 3DSSG 的下载链接。

为了使这些数据能够被模型有效利用，我们需要对其进行预处理，以获得坐标对齐的标注场景示例分割文件。首先，本工作从 3DSSG 数据集中提取了原始的点云数据。这些数据通常以 PLY 文件格式存储，其中包含了点云的三维坐标（x，y，z）。然而，由于数据采集过程中的误差以及不同设备之间的差异，这些原始数据可能存在一定程度的不一致和偏差。因此，在进行后续分析之前，需要对这些数据进行校准和变换。

表 5　3DSSG 数据集组织形式

文件名	功能
labels.instances.annotated.ply	场景的实例分割演示，包含点云和实例分割标签
mesh.refined.0.010000.segs.json	存储 obj 文件下每个点所属的 segment 序号
mesh.refined.mtl	网格的材质信息，包括光照参数和纹理信息
mesh.refined.obj	优化后（压缩）的点云文件，包含每个点的坐标、法向量、纹理坐标和三角面片信息
semseg.json	每个扫描的分割集以及物体的方向包围盒信息
sequence.zip	用于重建的图像序列
classes.txt	包含物体所有可能所属的类别
relationships.txt	包含谓词所有可能所属的类别
relationships_train.json	包含用于训练的场景扫描，以及扫描中的所有物体和关系的地面真实值
relationships_validation.json	包含用于验证的场景扫描，以及扫描中的所有物体和关系的地面真实值

为了实现数据的校准和变换，我们首先需要获取点云数据的变换矩阵。这些矩阵可以从数据集中的 Scan3R.json 文件中提取。变换矩阵是一个 4×4 的矩阵，包含了旋转、平移和缩放等空间变换信息。通过将点云数据与这些变换矩阵相乘，我们可以将数据从原始坐标系转换到一个统一的参考坐标系。本工作编写脚本读取了输入的 PLY 文件，从中提取点云数据，然后将点云数据扩展为齐次坐标形式，即在每个点的末尾添加一个值为 1 的元素。之后，将点云数据与对应的变换矩阵相乘，从而实现空间变换。为了高效地处理大量的点云数据，本文使用了多线程技术，根据计算资源的实际情况灵活调整并行处理的任务数量，显著提高了数据预处理的速度，缩短整个实验周期。

完成数据预处理后，我们获得了一组统一坐标系下的点云数据。这些数据已经经过校准和变换，因此可以直接用于模型的训练和测试。图 6 给出 3DSSG 四个场景扫描的示例，包括原始的 3D 点云和处理后的标注重建。

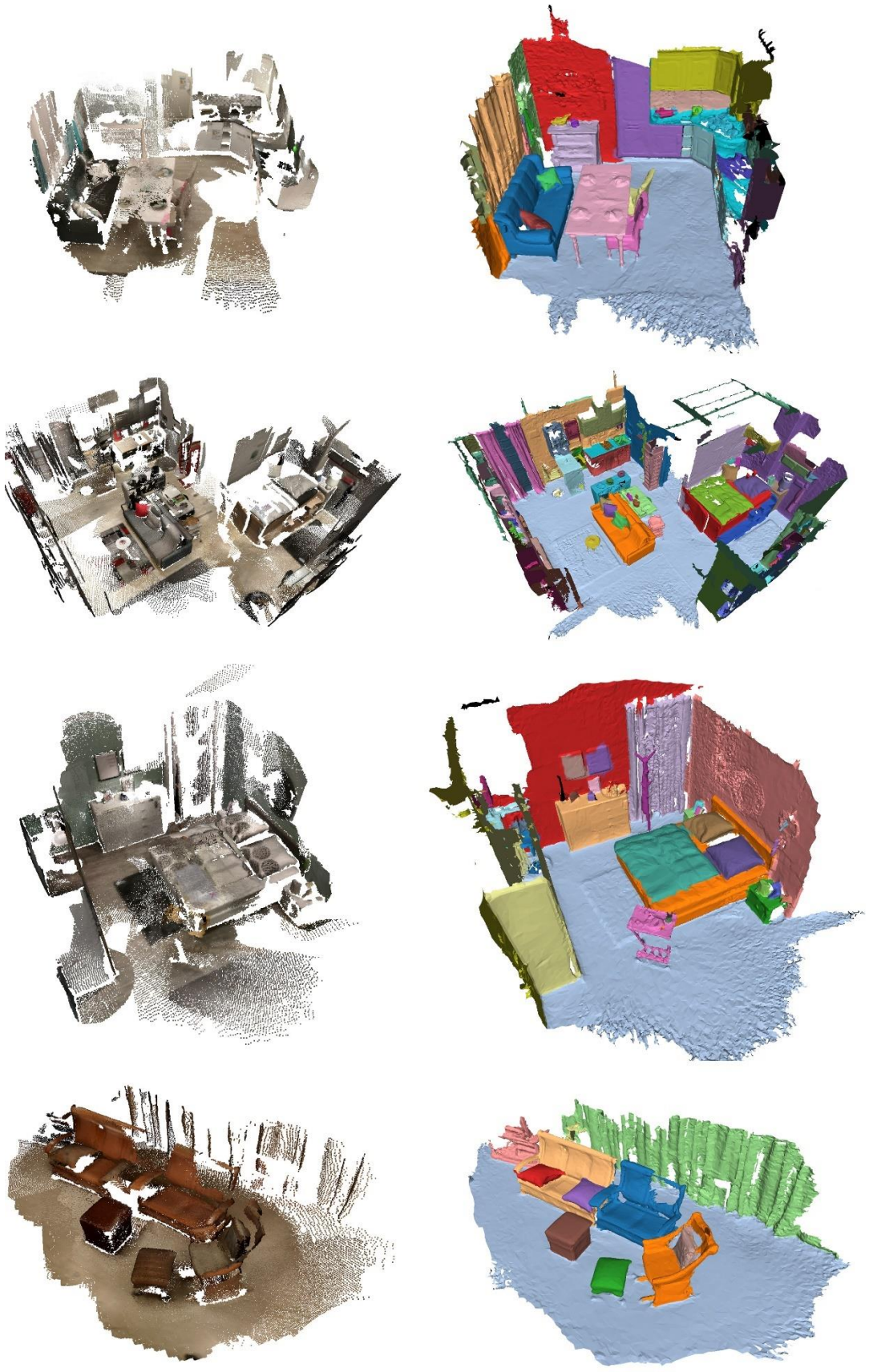

图 6 原始 3D 点云（左）和坐标对齐的语义标注的场景分割重建（右）。

4.5 评估指标

在本节中，本文将详细讨论场景图生成模型性能评估的关键指标：top-k 召回率。该指标用于量化模型在预测任务中的表现，以便于对模型进行改进和优化。在场景图生成任务中，从候选实体和关系中选择最相关的标签具有一定的挑战性。top-k 召回率用于评估模型预测的 top-k 个标签与真实标签之间的匹配程度。具体来说，它衡量了模型在 top-k 预测标签中正确预测的标签数量占真实标签总数的比例。数学上，top-k 召回率 $R@k$ 可以表示为：

$$R@k = \frac{\sum_{y \in Y} \mathbb{1}\left(\hat{Y_k} \cap y\right)}{|Y|}. \tag{16}$$

在这个公式中，$\hat{Y}$是预测的标签集合，Y 是真实的标签集合。$\hat{Y_k}$表示预测的 top-k 标签集合。$\mathbb{1}(A)$ 是指示函数，当条件 A 成立时，$\mathbb{1}(A) = 1$，否则 $\mathbb{1}(A) = 0$。

top-k 召回率在场景图生成任务中具有多方面的优势。首先，计算 top-k 召回率可以帮助我们量化模型在预测 top-k 标签时的表现，从而为模型的改进和优化提供参考。其次，不同模型在预测实体和关系时可能具有不同的性能。通过比较各个模型的 top-k 召回率，可以确定在生成场景图任务中哪个模型具有更好的性能。此外，在模型训练过程中，可以利用 top-k 召回率来监控模型的性能。根据 top-k 召回率的变化趋势，可以对模型参数进行适当调整以优化预测结果。top-k 召回率作为场景图生成模型的评价指标，有助于全面了解模型在预测任务中的表现，为模型优化和改进提供依据。

在场景图预测任务中，本文首先独立地对物体和谓词做 top-k 召回率评估。对于整体场景图的预测性能，和以往的工作[20][21][22][23]一样，本工作采取联合评估的方式，通过乘以各自的独立预测得到的 top-k 召回率分数得到排序的三元组分类分数列表。对于物体的评估，表 6 中给出 R@5 和 R@10 的结果；对于谓词的评估，表 6 中给出 R@3 和 R@5 的结果；对于关系的评估，表 6 中给出 R@50 和 R@100 的结果。结果越大表示模型性能越好。

4.6 语义场景图预测

本节将介绍使用 ASGPN 模型及基线模型在 3DSSG 数据集上进行语义场景图预测的实验结果。本文将比较这些模型在预测性能上的差异，并分析实验结果以阐述 ASGPN 模型在场景图生成任务中的优势。

本文依次对 ASGPN 以及 4.2 中提到的 6 个基线模型分别进行了训练和验证。在超参数方面，我固定了最大训练迭代次数为 400，学习率为 0.001。ASGPN 使用了两层 3.2.2 节中的逐特征注意力网络层，注意力头数为 8。物体特征、边特征以及一切的隐藏层特征维度都被设置为 512。设置 $\alpha = 0.1$ 和 $\beta = 1.0$ 计算 (17) 式的损失值。本工作控制了所有实验采用相同的随机种子和训练集验证集分割。对于每一次的训练和验证，在 4.1 节中的实验环境需要运行大约 11 个小时。

训练过程中，本工作使用 tensorboard 监控了模型的损失值及 top-k 召回率等评价指标，以确保模型正常收敛。经过训练，ASGPN 模型及基线模型在 3DSSG 数据集进行了验证。本文在表 6 中对比了这些模型在不同预测任务上的性能，包括物体预测、谓词预测以及整体场景图预测。实验结果表明，ASGPN 模型在各个预测任务上均优于基线模型，尤其是展现出了比当前先进的 3D 场景图生成模型更好的性能。具体来说，ASGPN 相比原始 SGPN 平均提升了 19.68%，相比 EdgeGCN 平均提升了 10.75%。

如表 6 结果所示，缺少 GNN 的初始特征仅能表现语义图较为粗糙的结构。而独立更新节点特征和边特征的 GCN 和 GAT 虽然在谓词预测方面有所优化，却在节点更新方面产生了严重的信息更新不准确，这可能是局部特征信息的损失或者过度平滑问题的原因导致的。相反，GraphSAGE 表现出了较好的预测能力，甚至优于基础的 SGPN 模型。这主要是因为 GraphSAGE 采用采样和聚合的策略来学习节点表示，这使得它在处理大规模图数据时具有较好的性能，同时有助于减轻过度平滑问题。

作为 3D 场景图生成的代表模型，SGPN 通过使用三元组图卷积网络和残差连接，使模型在 PointNet 提取到的初始特征基础上学习到了更多的拓扑信息。而其改进版本 EdgeGCN 更加关注边的信息，通过利用节点和边的孪生关系进一步探索特征嵌入的学习过程，使谓词预测的能力明显提升。

本文提出的 ASGPN 算法在物体、谓词和关系的预测上都表现出了强大的语义建模能力。一方面，这得益于多头逐特征注意力机制的学习能力，它从不同的角度关注节点和边特征，从而更好地捕获场景图中的多样性和复杂性。另一方面，ASGPN 引入了类似 EdgeGCN 的边注意力掩码来增强模型对边信息的关注。通过边注意力掩码，ASGPN 能够在学习节点特征的同时，更加关注邻近节点间的联系。这有助于更好地挖掘实体间的关系信息，从而提高场景图生成任务的性能。

表 6　场景图预测评估

方法	关系		物体		谓词	
	R@50	R@100	R@5	R@10	R@3	R@5
w/o GNN	0.39	0.45	0.66	0.77	0.62	0.88
GCN [27]	0.15	0.22	0.27	0.42	0.92	0.97
GraphSAGE [28]	0.55	0.63	0.65	0.77	0.93	0.97
GAT [29]	0.37	0.45	0.45	0.58	0.92	0.96
SGPN [23]	0.40	0.66	0.68	0.78	0.89	0.93
EdgeGCN [21]	0.49	**0.77**	0.66	0.80	**0.94**	**0.98**
ASGPN	**0.72**	**0.77**	**0.75**	**0.84**	0.93	**0.98**

图 7 给出了 GraphSAGE、SGPN 和 ASGPN 的可视化示例。本文展示了哈希 ID 为 20c993b3-698f-29c5-859c-dca8ddecf220 和 2e369567-e133-204c-909a-c5da44bb58df 的两个场景扫描，使用上述三种不同模型产生的预测结果。为了更直观简洁的展示，可视化展示去除了部分节点和关系。其中，观察实验结果发现 SGPN 方法在节点预测任务中表现不佳，性能值得改善。GraphSAGE 相比 SGPN 有了一定提升。而 ASGPN 同时在节点预测和边预测中都表现良好，基本能够正确构建 3D 场景图。

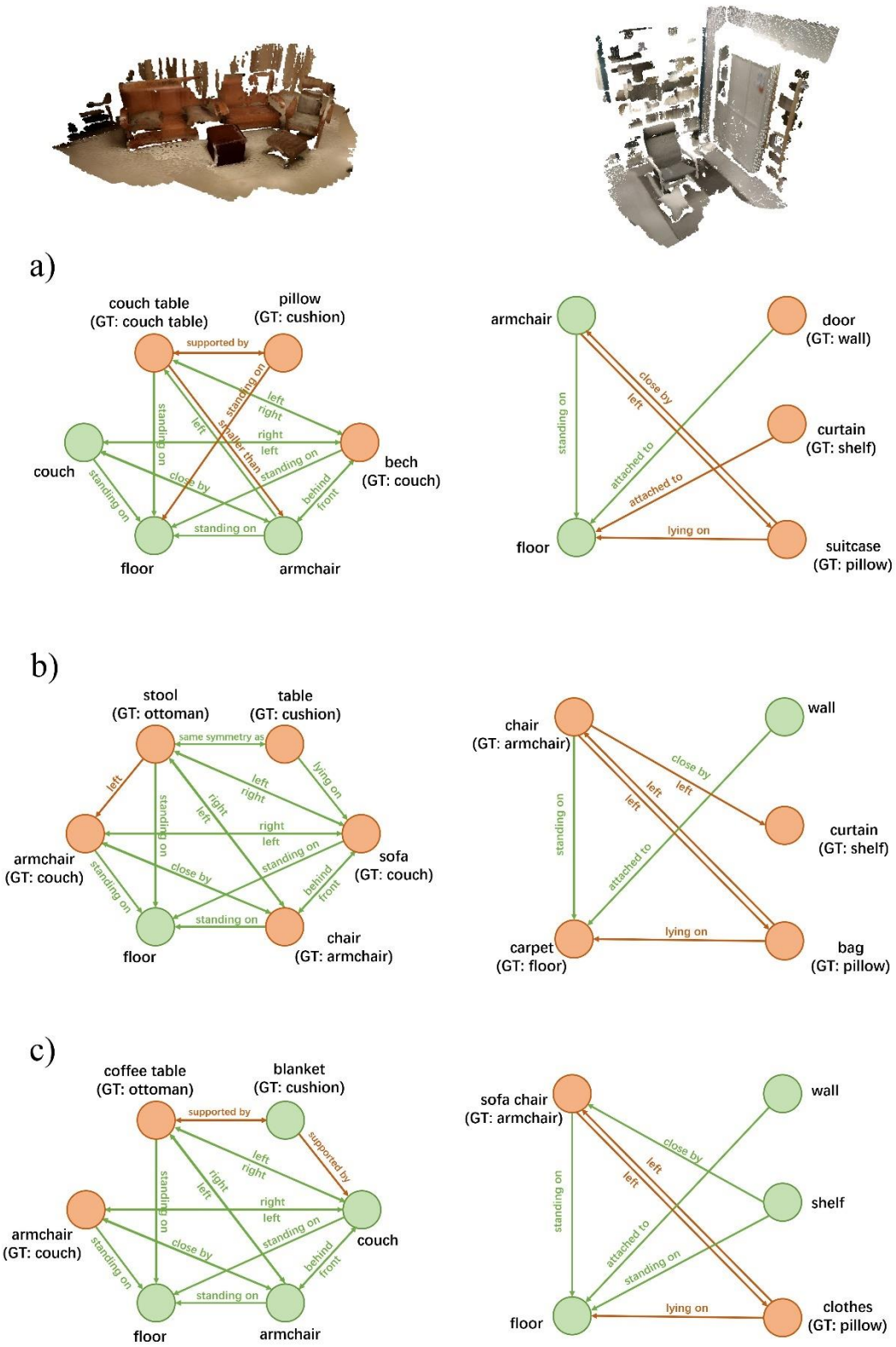

图 7 两个场景图预测的可视化，分别使用 a) GraphSAGE，b) SGPN 和 c) ASGPN 的预测结果。绿色的节点和边表示正确预测，橙色的节点和边表示错误预测，GT 表示地面真实值。

4.7 消融实验

本文提出了一种创新性的 3D 场景图生成算法 ASGPN，其中包含节点逐特征注意力模块和边注意力掩码模块。为了验证这两个模块对模型性能的贡献，本文进行了消融实验。本节将详细介绍消融实验的设置、过程和结果。

消融实验的目的是通过移除模型中的某个模块，观察模型性能的变化，从而了解该模块的作用和贡献。本研究分别设计了四种不同的实验设置：

1) **基线模型（baseline）：**不包含节点逐特征注意力和边注意力掩码的原始模型，和 4.2 节中的 w/o GNN 一致，作为对照组；

2) **只包含节点注意力的模型（w/o E）：**在基线模型基础上添加节点逐特征注意力机制；

3) **只包含边注意力的模型（w/o N）：**在基线模型基础上添加边注意力掩码；

4) **完整的 ASGPN 模型：**同时包含节点逐特征注意力和边注意力掩码的完整模型。

本节依旧使用 3DSSG 数据集作为实验数据集，并采用标准的评估指标 top-*k* 召回率来衡量模型性能。通过对比不同实验设置下的结果，可以分析节点逐特征注意力和边注意力掩码的作用和贡献。本节的实验设置遵循了 4.6 节中的超参数设置，并控制了所有实验采用相同的随机种子和训练集验证集分割。

表 7　消融实验评估

方法	关系		物体		谓词	
	R@50	R@100	R@5	R@10	R@3	R@5
baseline	0.39	0.45	0.66	0.77	0.62	0.88
w/o E	0.70	0.76	0.73	0.82	0.93	0.97
w/o N	0.53	0.61	0.61	0.74	**0.94**	**0.99**
ASGPN	**0.72**	**0.77**	**0.75**	**0.84**	0.93	0.98

实验结果表明，相对于基线模型，只包含节点逐特征注意力的模型的 top-*k* 召回率平均提升了 37.52%，并且在各项评估指标上均有显著提升。这说明节点逐特征注意力机制能够有效地捕捉节点特征中的局部和全局信息，从而提高模型对场景结构的理解。具体来说，节点逐特征注意力通过对不同节点特征进行加权求和，使得模型能够关注到具有更高语义信息的特征，从而提高场景图生成的准确性。

同样地，与基线模型相比，只包含边特征的模型的 top-*k* 召回率平均提升了 20.68%，在多数评估指标上也有提高。在谓词关系上的提升尤为明显。这表明边注意力掩码在捕捉物体间关系方面起到了关键作用。边注意力掩码通过计算演化后节点的注意力权重，为物体间的关系提供了更精细的表示，使得模型能够更好地理解场景中物体的相互作用和联系。

最后，当将两种创新的注意力机制同时应用于 ASGPN 模型时，实验结果显示模型在各项评估指标上均取得了最优表现。ASGPN 相比不使用图神经网络的基线模型的 top-*k* 召回率平均提升了 39.97%。这证明了节点逐特征注意力和边注意力掩码在模型中具有互补作用，

从而进一步提升模型性能。相较于基线模型，完整的 ASGPN 模型在场景图生成任务中表现出了更高的准确性、更强的泛化能力和更佳的稳定性，在处理复杂场景、捕捉细节特征和理解物体间关系方面具有更强的能力。这些实验结果进一步证实了本工作的设计思路和所提出的创新模块的有效性。通过引入这两个创新性模块，本文提出的模型能够更好地捕捉场景中的节点特征和物体间关系，从而在 3D 场景图生成任务中取得了优越的性能。

5 总结与展望

5.1 优点与局限性

本文针对复杂且多样的 3D 场景，提出了一种新颖且有效的 3D 场景图生成模型——注意力场景图预测网络（ASGPN）。与现有的基于 3D 点云的场景图生成算法相比（如 SGPN），ASGPN 具有以下优点：1）通过引入逐特征注意力机制，ASGPN 能够更好地捕捉物体和关系之间的权重信息，使网络在特征交互中更加关注关键的特征信息。这有助于提高模型对复杂 3D 场景中关键视觉线索的捕捉能力。2）针对关系特征，本文提出了一种利用节点特征的注意力掩码，以此协助关系特征在多层感知机中更精确地演化。与 SGPN 原本的图卷积网络部分相比，ASGPN 的优化注意力图卷积部分在关系特征交互上有明显的进步。

然而，ASGPN 仍然存在一定的局限性：1）在处理形态各异的 3D 对象实例时，ASGPN 可能仍然难以完全捕捉所有的视觉信息，这点在模型对物体识别的相对低精准度上有所体现。2）虽然 ASGPN 在 3DSSG 数据集上取得了显著的性能提升，但 3DSSG 几乎是目前唯一可用的 3D 语义场景图数据集，其泛化能力仍有待进一步验证。在未来的工作中，需要在更多数据集上对 ASGPN 进行验证，以证明其广泛适用性。

5.2 未来工作展望

基于 ASGPN 的优点与局限性，本文提出以下未来工作展望：1）改进注意力机制：虽然本文提出的逐特征注意力机制和注意力掩码在某种程度上提高了模型的性能，但仍有优化空间。未来可以尝试研究更加强大且灵活的注意力机制，以进一步提高模型在捕捉关键视觉线索方面的能力。2）探索多模态输入：为了提高模型的泛化能力，可以考虑将多模态输入（如 RGB 图像、深度图像、先验知识等）融入 ASGPN，使模型能够利用多种类型的数据来提高对 3D 场景的理解。3）适应动态场景：现有的 ASGPN 主要针对静态 3D 场景进行场景图生成。在未来的工作中，可以尝试将 ASGPN 扩展到动态场景中，以实现对动态变化的场景元素的实时建模和分析。4）自适应场景图生成：当前 3DSSG 数据集应用场景主要考虑室内场景，更加通用化的模型应当考虑到不同类型场景下物体和关系的复杂程度存在差异，未来可以研究一种自适应的场景图生成方法。通过根据输入场景的复杂性和特点自动调整模型结构和参数，以实现在不同场景下都能取得良好性能的场景图生成。

【参考文献】

[1] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436-444.

[2] Schuster S, Krishna R, Chang A, et al. Generating semantically precise scene graphs from textual descriptions for improved image retrieval[C]//Proceedings of the fourth workshop on vision and language. 2015: 70-80.

[3] Liang X, Lee L, Xing E P. Deep variation-structured reinforcement learning for visual relationship and attribute detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 848-857.

[4] Chang X, Ren P, Xu P, et al. A comprehensive survey of scene graphs: Generation and application[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 45(1): 1-26.

[5] Armeni I, He Z Y, Gwak J Y, et al. 3d scene graph: A structure for unified semantics, 3d space, and camera[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 5664-5673.

[6] Yang Y, Zhuang Y, Pan Y. Multiple knowledge representation for big data artificial intelligence: framework, applications, and case studies[J]. Frontiers of Information Technology & Electronic Engineering, 2021, 22(12): 1551-1558.

[7] Johnson J, Krishna R, Stark M, et al. Image retrieval using scene graphs[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3668-3678.

[8] Hou J, Dai A, Nießner M. 3d-sis: 3d semantic instance segmentation of rgb-d scans[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 4421-4430.

[9] Lahoud J, Ghanem B, Pollefeys M, et al. 3d instance segmentation via multi-task metric learning[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 9256-9266.

[10] 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.

[11] 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.

[12] 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.

[13] Zhou Y, Tuzel O. Voxelnet: End-to-end learning for point cloud based 3d object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 4490-4499.

[14] Krishna R, Zhu Y, Groth O, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations[J]. International journal of computer vision, 2017, 123: 32-73.

[15] Xu D, Zhu Y, Choy C B, et al. Scene graph generation by iterative message passing[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 5410-5419.

[16] Qi M, Li W, Yang Z, et al. Attentive relational networks for mapping images to scene graphs[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 3957-3966.

[17] Yang J, Lu J, Lee S, et al. Graph r-cnn for scene graph generation[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 670-685.

[18] Armeni I, He Z Y, Gwak J Y, et al. 3d scene graph: A structure for unified semantics, 3d space, and camera[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 5664-5673.

[19] Kim U H, Park J M, Song T J, et al. 3-D scene graph: A sparse and semantic representation of physical environments for intelligent agents[J]. IEEE transactions on cybernetics, 2019, 50(12): 4921-4933.

[20] Zhang S, Hao A, Qin H. Knowledge-inspired 3d scene graph prediction in point cloud[J]. Advances in Neural

Information Processing Systems, 2021, 34: 18620-18632.

[21] Zhang C, Yu J, Song Y, et al. Exploiting edge-oriented reasoning for 3d point-based scene graph analysis[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 9705-9715.

[22] Wu S C, Wald J, Tateno K, et al. Scenegraphfusion: Incremental 3d scene graph prediction from rgb-d sequences[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 7515-7525.

[23] Wald J, Dhamo H, Navab N, et al. Learning 3d semantic scene graphs from 3d indoor reconstructions[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 3961-3970.

[24] Rusu R B, Cousins S. 3d is here: Point cloud library (pcl)[C]//2011 IEEE international conference on robotics and automation. IEEE, 2011: 1-4.

[25] 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.

[26] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model[J]. IEEE transactions on neural networks, 2008, 20(1): 61-80.

[27] Kipf T N, Welling M. Semi-Supervised Classification with Graph Convolutional Networks[C]//International Conference on Learning Representations, 2017.

[28] Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[J]. Advances in neural information processing systems, 2017, 30.

[29] Velickovic P, Cucurull G, Casanova A, et al. Graph attention networks[J]. stat, 2017, 1050(20): 10.48550.

[30] Xu K, Hu W, Leskovec J, Jegelka S. How Powerful are Graph Neural Networks?[C]//International Conference on Learning Representations, 2019.

[31] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[C]//International Conference on Learning Representations, 2015.

[32] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]. Advances in neural information processing systems, 2017, 30.

[33] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

[34] Wang T, Jin D, Wang R, et al. Powerful graph convolutional networks with adaptive propagation mechanism for homophily and heterophily[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2022, 36(4): 4210-4218.

[35] ShunChengWu. 3DSSG[Source code]. 2021. https://github.com/ShunChengWu/3DSSG/tree/master

[36] Wald J, Avetisyan A, Navab N, et al. Rio: 3d object instance re-localization in changing indoor environments[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 7658-7667.

致谢

衷心感谢王旭老师在本次毕业设计中所提供的选题、学术指导、论文修改建议和服务器算力支持，同时也感激您为我大学生涯提供的宝贵的实习机会和升学推荐。感谢邢炜老师在我的科研道路上的启蒙，为我提供了难得的科研机会、学术指导和升学推荐。在此也要表达我对冯禹洪老师在我修读“人工智能+”微专业时期的悉心指导、大二转专业时期的大量帮助以及提供的升学推荐的感激之情。同时，感谢梁正平老师对我担任课程课代表的信任和提供的升学推荐。最后，我要感激 Akeel Shah 教授为我的学术研究论文提供修改建议和其他指导、帮助。

我还要感谢李逸凡师兄为本次毕设工作提供的大量指导和讨论。在此，也要感激白虎群、上岸群、密室群、老人群的所有群友，以及 AutoLeaders 俱乐部、机电学生会宣传部、机电足球队的各位同学在本科生涯的陪伴。同时也深深感谢支持我未来升学的父母和朋友们。

感谢深圳大学为我提供了转专业的可能性，以及四年来的奖学金、实验室设备和课程支持。我要感激埃默里大学的赏识和奖学金资助。感谢开源社区为我和其他研究者提供的无私奉献，使我们能在深度学习的科研道路上不断前进。

最后，感谢四年来努力钻研、熬夜奋斗、不断坚持的自己。这段时间的经历将一个贪玩的小孩改造成了对未来学术道路充满好奇与信心的机器学习研究者。希望未来的我能不负这四年的努力和拼搏，继续茁壮、成长。

再次向所有给予我帮助、支持和关心的老师、同学、家人和朋友们表示衷心的感谢。

Research on Attention-based 3D Scene Graph Generation Algorithm

【**Abstract**】Scene Graph is a structured representation of a scene that expresses the objects, attributes, and relationships between objects. Similar to generating a Scene Graph from 2D images, the goal of the 3D Scene Graph generation is to construct a representation that provides numerically accurate quantification of the object relationships in 3D scenes. Scene Graph Prediction Network (SGPN), which is considered as the most advanced 3D scene graph generation framework at present, uses Graph Neural Network (GNN) to learn the features of objects and relations in the scene graph. Despite its effectiveness, the propagation strategy of GNN can hardly capture visual clues in complex 3D scenes. Based on SGPN, this paper proposes an attention-based optimization strategy for the 3D Scene Graph generation: 1) for node features, we introduce a feature-wise attention mechanism to capture weight information between objects and relationships; 2) for relationship features, we introduce an attention matrix that utilizes node features to assist in the evolution of relationship features. Experimental results show that my proposed model achieves a recall rate improvement of 19.68% over SGPN on the 3DSSG dataset with 1482 scene graphs. Further ablation experiments and visualization demonstrations confirm the effectiveness and robustness of the optimized model.

【**keywords**】 3D scene; Scene Graph Generation; Graph Neural Network; Attention

指导教师：王旭