

## 数据驱动的智能 CAD 研究综述

谭柏君, 徐延宁\*, 李曼祎, 王璐

(山东大学软件学院 济南 250101)  
(xyn@sdu.edu.cn)

**摘要:** 传统 CAD 系统存在效率低下、创新受限与设计周期长等不足, 相比之下智能 CAD 技术有自动化、智能化和具备处理复杂任务能力等优势. 综述了 AI 与 CAD 结合的相关工作, 首先介绍 CAD 领域中常用的数据集; 然后从 CAD 表示、正向工程和逆向工程 3 个方面概述 AI 对 CAD 流程的改变; 再对近年来迅速发展的大语言模型对 CAD 的推动作用进行探讨; 最后对智能 CAD 的研究现状进行总结, 并探讨了高质量数据集构建、生成精度与处理规模的提升、大语言模型深度集成, 以及 CAD 与 CAE 协同优化等未来可能的发展趋势.

**关键词:** 智能 CAD; 数据驱动; CAD 重建; 大语言模型

**中图分类号:** TP391.41 **DOI:** 10.3724/SP.J.1089.2025-00241

## A Review of Data-Driven Intelligent CAD Research

Tan Baijun, Xu Yanning\*, Li Manyi, and Wang Lu

(School of Software, Shandong University, Jinan 250101)

**Abstract:** Traditional CAD systems have shortcomings such as low efficiency, limited innovation, and long design cycles. In contrast, intelligent CAD technology has advantages such as automation, intelligence, and the ability to handle complex tasks. This article provides an overview of the related work on the integration of AI and CAD. Firstly, it introduces commonly used datasets in the CAD field. Then, it summarizes the changes that artificial intelligence has brought to the CAD design process from three aspects: CAD representation, forward engineering, and reverse engineering. Further, it explores the driving role of the rapidly developing Large Language Model in CAD design in recent years. Finally, the research status of intelligent CAD was summarized, and potential future development trends such as high-quality dataset construction, improvement of generation accuracy and processing scale, deep integration of Large Language Models (LLM), and collaborative optimization of CAD and CAE were discussed.

**Key words:** intelligent CAD; data driven; CAD reconstruction; large language model

CAD 作为现代工程设计、建筑规划和制造业中的核心工具, 其任务涵盖了几何建模、参数化设计、装配约束求解、逆向工程、仿真分析等多个方面. 传统的 CAD 系统主要依赖明确的几何描述、

参数化建模和规则驱动的算法, 通过工程师的手工操作和经验积累实现设计目标. 随着工程复杂度的不断提高和产品个性化需求的激增, 传统方法在自动化、智能化和高效性方面逐渐暴露出局限

收稿日期: 2025-07-03; 修回日期: 2025-08-25. 基金项目: 国家重点研发计划(2022YFB3303203); 国家自然科学基金(62272275). 谭柏君(2002—), 男, 硕士研究生, CCF 学生会员, 主要研究方向为 CAD&CG; 徐延宁(1975—), 男, 博士, 副教授, 硕士生导师, CCF 会员, 论文通信作者, 主要研究方向为计算机图形学; 李曼祎(1992—), 女, 博士, 副研究员, 硕士生导师, CCF 会员, 主要研究方向为计算机图形学、计算机视觉、人工智能; 王璐(1981—), 女, 博士, 教授, 博士生导师, CCF 会员, 主要研究方向为计算机图形学、三维建模、真实感渲染.

性. 例如, 传统方法在面对复杂设计任务时往往效率低下且容易出错, 尤其是在处理大规模装配体、自由曲面建模和多目标优化问题时, 传统 CAD 系统的局限性尤为明显; 传统 CAD 系统在处理非结构化数据(如点云、网格)时缺乏自动化和智能化的工具, 导致设计迭代周期长, 难以满足现代工业设计对快速原型设计和高效优化的需求.

近十年来, 人工智能(artificial intelligence, AI)技术取得了突破性的进展, 其革新浪潮也深刻地影响了 CAD 领域的发展. 以深度学习、生成模型和大语言模型(large language model, LLM)为代表的技术突破, 推动着 CAD 从工具驱动向知识与数据驱动转变. 图神经网络(graph neural network, GNN)<sup>[1]</sup>及其变体能有效地提取 CAD 模型数据的几何和拓扑信息, 为数据的进一步处理提供了有力支撑; 生成对抗网络(generative adversarial networks, GANs)<sup>[2]</sup>和变分自编码器(variational autoencoders, VAEs)<sup>[3]</sup>等生成模型因其强大的数据驱动建模能力而被用于 CAD 建模生成相关的任务; Transformer 架构<sup>[4]</sup>能够处理文本、语音等序列数据, 适用于自然语言处理的相关任务, 这一特性也催生出 CAD 模型的命令序列表示方式<sup>[5]</sup>. 此外, 基于 Transformer 架构的大规模预训练模型(如 GPT 系列、DeepSeek 等)<sup>[6-11]</sup>通过理解自然语言指令, 使设计流程脱离了传统参数化操作的框架, 用户可通过对话式交互实时生成三维模型数据. 目前, 智能 CAD 已在多个工业和设计领域取得广泛应用<sup>[12-16]</sup>, 并呈现出流程自动化、知识专业化、系统集成化等特点<sup>[17]</sup>.

深度学习的模式识别能力和逻辑推理能力使得 CAD 系统能够理解数据背后的工程语义并进行分析决策; 同时, 数据驱动的生成式设计也已经覆盖了从需求输入到方案迭代的完整 CAD 流程, 如 Autodesk 把 CAD 过程视为一个序列决策过程, 应用神经引导搜索等方法重建或优化 CAD 程序<sup>[18]</sup>. 未来, 随着 AI 技术的进一步发展, CAD 系统将更加智能化和自动化, 能够处理更复杂的设计任务, 并在跨领域协同设计、参数化优化和语义驱动建模等方面发挥更大的作用.

## 1 研究现状

智能 CAD 正在经历由传统经验驱动向数据驱动的转变, 在深度学习的知识框架下, 许多关于 CAD 数据的研究也随之诞生, 其中有 2 类研究方

向值得关注. (1) 数据集的构建与生成, 为数据加工和学习提供数据支撑, 是基础性的研究任务; (2) CAD 数据的表示作为智能 CAD 的上游任务, 需要为神经网络构建可学习的 CAD 表示形式, 也是重要的研究方向之一. 在实际应用中, 根据 CAD 中设计流程的不同需求又可以分成正向工程和逆向工程 2 类研究方向, 其中, 正向工程侧重于从概念设计到建模优化的通用设计流程, 而逆向工程则聚焦于利用现有数据进行 CAD 重建. 近年来, LLM 的崛起对生成式设计的数据产出模式产生了深远影响, 涌现出越来越多的将 LLM 与 CAD 技术相融合的创新工作, 成为了新的研究热点. 研究现状总览如图 1 所示.



图 1 智能 CAD 研究现状

(1) 数据集. 数据集作为智能 CAD 研究的重要基石, 其质量与规模直接影响算法模型的性能和泛化能力<sup>[19]</sup>. 近年来, 针对智能 CAD 系统的各类需求, 学术界构建了涵盖二维草图<sup>[20]</sup>、三维几何模型<sup>[21]</sup>、B-Rep 数据<sup>[22]</sup>、装配结构<sup>[23]</sup>、文本描述<sup>[24]</sup>等多模态数据集. 一方面, 部分数据集来源于真实工业设计和制造流程, 其数据质量和标注精度较高, 能够真实地呈现工程实践中的设计要求<sup>[25]</sup>; 另一方面, 为了弥补真实数据采集成本高昂与样本不足的问题, 研究者们采用合成数据生成技术制作出可控性强的虚拟数据集<sup>[26]</sup>. 目前已有不少关于 CAD 数据集的工作, 但受制于标注数据所需的强专业知识以及行业中的专利壁垒, 高质量的数据集依然十分稀缺, 特别是缺少带有完整建模序列的 CAD 数据集.

(2) CAD 表示. CAD 表示是智能 CAD 系统的核心基础, 决定了 CAD 数据的存储、处理和生成方式. 近年来, 随着深度学习技术的不断发展, 基于深度学习的 CAD 数据表示方法逐渐成为主流. 根据处理问题的不同, 可以细分为 CAD 模型的神

经表示<sup>[27]</sup>、B-Rep 表示<sup>[28]</sup>和装配建模<sup>[23]</sup>这 3 个子领域。通过神经网络的强大表达能力, CAD 表示方法能够学习几何体的结构与内在关系, 提高学习和存储效率, 并支持更加复杂的建模和自动化设计任务。在此过程中, CAD 模型的表示不仅限于传统的几何信息, 还涉及拓扑结构、约束条件等多维度数据, 为正向工程和逆向工程的实现提供了坚实的基础, 是整个智能设计流程中的关键环节。

(3) 正向工程. 正向工程是 CAD 的基础流程, 主要任务是根据初始设计需求逐步构建出完整的几何模型; 核心任务包括草图生成<sup>[29]</sup>和基于草图的三维建模<sup>[26]</sup>, 通过智能化工具快速地完成设计迭代。在智能 CAD 中, AI 技术的应用极大地加速设计迭代, 并提升了设计的自动化水平。使用基于深度学习的算法, AI 能够实现自动化绘图与建模, 减轻设计师在草图阶段的重复性劳动, 并且通过参数化设计, 设计师可以灵活地调整模型结构, 满足不断变化的设计需求。

(4) 逆向工程. 逆向工程从已有的物理或数字模型(如点云、体素数据等)出发, 通过分析和重建恢复 CAD 模型的几何结构数据, 提取设计信息, 是设计优化和改进的重要手段。本文根据不同的实现路线, 将逆向工程的研究分为草图拉伸法<sup>[30]</sup>、构造实体几何(constructive solid geometry, CSG)法<sup>[31]</sup>和分割拟合法<sup>[32]</sup>这 3 类, 每类有不同的数据表示

方式。在智能 CAD 领域, AI 技术的引入使得逆向工程的过程大幅度加速, 尤其是在对复杂几何体进行自动化分割、拟合与重建时, AI 模型通过学习大量的几何特征数据, 能够快速、准确地提取出结构信息, 显著减少了手工建模的时间和工作量。

(5) LLM&CAD. LLM 与 CAD 的结合, 为智能设计带来了全新的交互方式。通过自然语言处理, 设计师可以通过直接的语义描述与 CAD 系统进行交互, 实现从语言到几何模型的自动转换<sup>[33]</sup>。这一进展不仅提升设计的便捷性, 也极大地降低了设计人员对复杂操作的学习门槛, 尤其在生成初步设计草图或自动化调整现有模型时, 能够提供有效的帮助。目前, 尽管 LLM 在 CAD 领域的应用仍处于初步阶段, 研究较为匮乏, 但是随着自然语言处理技术的不断成熟, LLM 有望在 CAD 识别、自动生成及优化等任务中发挥巨大的潜力, 成为未来智能 CAD 发展的重要方向之一。

## 2 数据集

在智能 CAD 领域, 数据集作为算法模型训练与验证的基础载体, 其质量与多样性直接决定了提取几何特征、智能生成、逆向重建等多种核心任务的性能边界。表 1 所示为常用的数据集信息。

表 1 常用 CAD 数据集信息

数据集名称	数量	主要表示方式	任务类型	描述
MCB <sup>[34]</sup>	58 696	网格	分类	带注释的机械部件数据集
Thing10K <sup>[21]</sup>	10 000	网格	分类	三维打印模型, 每个模型以三角网格形式存储
ShapeNet <sup>[35]</sup>	>3 000 000	网格	分类	每个三维模型带有丰富的语义注释
ModelNet <sup>[36]</sup>	151 128	网格	分类	经过人工数据清理的网格数据集
PartNet <sup>[37]</sup>	573 585	网格	分类、分割	适用于细粒度语义分割的三维部件标注数据集
Machining Feature <sup>[38]</sup>	23 995	网格	分类、分割	有 24 类加工特征的大规模合成数据集
FabWave <sup>[39]</sup>	5 373	B-Rep	分类	包含 52 个机械零件类型, 用 STEP 格式表示
SolidLetters <sup>[28]</sup>	96 000	B-Rep	分类	随机拉伸和填充 26 个字母(a~z)来生成
PVar <sup>[40]</sup>	120 000	B-Rep	分类、重建	用 SketchGraphs 的约束参数草图创建 B-Rep 模型
ABC <sup>[22]</sup>	>1 000 000	B-Rep	重建	机械模型, 支持参数化表面、曲线和拓扑结构
FURNITURE B-REP <sup>[41]</sup>	6 171	B-Rep	重建	包含规范类别标签和自由曲面的 B-Rep 数据集
MFCAD <sup>[42]</sup>	15 488	B-Rep	分割	自动生成的面向加工特征识别的数据集
MFCAD++ <sup>[43]</sup>	59 655	B-Rep	分割	用于特征识别, 每个模型包含 3~10 个相交特征, 更贴近真实复杂零件
Fusion 360 Segmentation <sup>[44]</sup>	35 858	B-Rep	分割、重建	包含面分割标签, 建模操作信息
ANSI Mechanical Component <sup>[32]</sup>	17 197	B-Rep	分割、重建	TraceParts 提供的 ANSI 机械部件 CAD 模型
CC3D-ops <sup>[25]</sup>	>37 000	B-Rep	分割、重建	更接近现实的工业模型, 功能和 Fusion 360 Segmentation 相似
ABCParts <sup>[45]</sup>	32 000	B-Rep	分割、重建	每个模型包含至少一个 B 样条曲面片
AutoMate <sup>[46]</sup>	>3 000 000	B-Rep	装配	B-Rep 类型的零件装配数据集
Fusion 360 Gallery assembly <sup>[23]</sup>	154 468	B-Rep	装配	提供几何、接触面和装配图信息, 用于无监督学习 CAD 部件自动装配

续表

数据集名称	数量	主要表示方式	任务类型	描述
DeepCAD <sup>[5]</sup>	178 238	命令序列	重建	模型表示为草图和拉伸操作的序列
Fusion 360 Reconstruction <sup>[18]</sup>	8 625	命令序列	重建	包含网格、B-Rep、命令序列等多种格式
Sketch2CAD <sup>[26]</sup>	>15 000 000	命令序列	重建	随机组合 CAD 操作序列生成三维模型及其对应的二维草图
Free2CAD <sup>[47]</sup>	>210 000	命令序列	重建	在 Sketct2CAD 的基础上添加笔画分组
SketchGraphs <sup>[20]</sup>	>15 000 000	命令序列	重建	大规模草图几何约束图数据集
CAD2Sketch <sup>[48]</sup>	6 000	命令序列	重建	基于 CAD 序列生成构造线与特征线, 优化可读性, 用于训练草图理解模型
OpenECAD <sup>[49]</sup>	>200 000	命令序列	重建	有完整的建模历史和编辑链路
CADParse <sup>[50]</sup>	40 000	命令序列	重建	包含五种操作类型的命令序列
ABC-mono <sup>[51]</sup>	208 853	命令序列	重建	成对渲染图像和命令序列表示 CAD 模型数据集
KOCAD <sup>[51]</sup>	100	命令序列	重建	成对真实图像和命令序列表示 CAD 模型数据集
CAD2Program <sup>[52]</sup>	368 000	参数化原语	重建	机柜模型和相应的二维工程图纸
CAD as Language <sup>[53]</sup>	>4 700 000	对象序列	重建	使用序列化协议缓冲区(protocol buffers, PB)表示, 包含草图的实体和约束
Img2CAD <sup>[54]</sup>	4 574	命令序列	重建	成对渲染图像、命令序列、语义标签的数据集
Text2CAD <sup>[24]</sup>	1 700 000	命令序列	重建	每个参数化 CAD 序列对应四层文本描述
Omni-CAD <sup>[55]</sup>	453 220	命令序列	重建	多模态数据, 包含建模命令序列和相应的文本描述、多视图图像和点云数据

与图像标注等常见的数据集标注任务不同, 构建高质量的 CAD 数据集需要大量的领域知识和专业经验, 从数据采集、预处理、标注到数据校验, 每个环节都必须紧密结合实际工程与设计需求。目前, 高质量的数据集依然是智能 CAD 进一步发展的一大瓶颈, 特别是缺少包含完整建模序列的数据集。构建高质量数据集的成本非常高昂, 需要投入大量人力、时间和资源, 同时还可能受到专利所有权和数据隐私等方面的限制, 使得数据的广泛收集和共享变得更加困难。这些挑战促使部分研究者探索合成数据集的路径。合成数据集可以通过模拟生成技术较低成本快速构建, 并在一定程度上弥补真实数据集不足的问题。

除了高质量外, 多样性也是智能 CAD 任务对数据集的一大需求。智能 CAD 因其丰富的任务类型, 特别是多模态任务, 需要图像数据(二维草图、渲染图), 几何数据(网格、B-Rep 模型), 结构化参数(设计树、约束关系)和自然语言标注(文本描述)等多种格式的数据作支撑, 目前, 多模态数据集有较大的空缺需要去填补。此外, 数据集的多样性不仅体现在数据的格式上, 还体现在数据来源、场景复杂性等因素上。例如, 大部分数据集都关注比较简单的机械 CAD, 目前还未见能关注到更复杂的电子 CAD 的研究, 也没有可供专门使用的数据集。如何在数据多样性和数据量之间达到平衡, 也成为当前研究的重要方向, 这需要不断地优化数据标注标准和增强数据处理自动化水平, 满足不断

扩展的智能 CAD 应用场景。同时, 由于工业设计数据高度依赖垂直领域(如汽车、航空航天、消费电子), 不同行业的设计标准, 文件格式(STEP, IGES, Parasolid 等)和语义标注体系差异显著, 难以构建统一的数据池, 如何在扩展数据多样性的同时保证数据表示方式的一致性也是一个难题。

CAD 数据集的发展过程是有迹可循的。早期的智能 CAD 任务以三维对象的分类<sup>[56]</sup>为主, 处理的数据对象主要是网格格式, 代表性的数据集有 MCB<sup>[34]</sup>, Thingi10K<sup>[21]</sup>, ShapeNet<sup>[35]</sup>, ModelNet<sup>[36]</sup>, PartNet<sup>[37]</sup>等。随着研究的深入, 研究任务变得多样化, 出现了如三维分割、逆向重建的应用等, 网格格式的 CAD 模型包含的信息已经不能满足这些任务的需求。B-Rep 是 CAD 领域常用的一种数据格式, 其包含曲线、曲面的参数和拓扑特征, 能够精准地表达 CAD 模型, 符合分割、重建等任务的数据需求。近年来, 各种不同用途、不同场景的 B-Rep 格式数据集陆续出现, 推动了智能 CAD 的发展。在这类数据集中, 影响较大、使用较广泛的是 ABC 数据集<sup>[22]</sup>。该数据集是一个专为几何深度学习设计的大规模 CAD 模型数据集, 包含超过 100 万个源自 Onshape 平台<sup>[57]</sup>的 B-Rep 模型, 涵盖机械零件、工业组件等人工设计对象; 其优势是参数化表示与精准标注: 模型以 NURBS 曲面、二次曲面等解析几何形式存储, 提供面片分解、锐利特征标注和微分属性(如法线、曲率)的精确真值。ABC 数据集为许多智能 CAD 工作提供了数据支撑, 也

有研究人员为了特定任务在该数据集的基础上进行了改进<sup>[45]</sup>. B-Rep 格式的 CAD 数据虽然包含足够的信息, 但是难以直接交给神经网络学习.

为了充分发挥神经网络的优势, 研究人员开始使用命令序列的形式表示 CAD 模型, 其中 2 个典型的工作 Fusion 360 Gallery<sup>[18]</sup>和 DeepCAD<sup>[5]</sup>. 两者均定义了一种简单、参数化、可学习的操作语言, 包括草图和拉伸 2 种操作, 并依照这 2 种操作描述模型. 在重建任务中, 将命令序列看作一种自然语言, 恢复命令序列和相应的参数即可恢复原来的模型. 得益于深度学习的发展和 Transformer 架构<sup>[4]</sup>对自然语言的强大处理能力, 这种重建方式被证明是可行的. 受到上述工作的启发, 研究人员开始重视命令序列的模型表达方式, 提出了许多新的方法, 分别适用于各个领域, 包括二维草图生成<sup>[20,49]</sup>、草图生成三维模型<sup>[26,48]</sup>、模型重建<sup>[50,52]</sup>等. 同时, 在 LLM 快速崛起的背景下, 一些如附带描述文本的适用于 LLM 的数据集也随之出现. Img2CAD<sup>[54]</sup>由 LLM 生成离散的命令序列; Text2CAD<sup>[24]</sup>使用视觉语言模型 (vision-language model, VLM) 为 CAD 模型生成文本描述, 并将其作为数据集的一部分用于生成模型的训练; OmniCAD<sup>[55]</sup>是一个多模态的数据集, 包含命令序列、文本、图像和点云多种模型表达形式, 为训练多模态 CAD 生成模型提供了必要的资源.

目前, 命令序列表示的数据集依然是稀缺的资源和研究的重点, 完整的命令序列不仅展示最终产品的构造, 更反映了设计过程中各个决策和步骤, 对于理解设计逻辑和优化自动化设计流程具有极高的价值. 由于专利限制、高领域知识需求等原因的限制, 要手动地构建一个高质量的命令序列数据集十分困难, 因此自动生成数据集就成为一个理想的方案: 一是自动生成的方式能够快速产生大量的模型数据, 规避专利限制的问题; 二是神经网络更适合处理命令序列的表示方式, 有能力生成高质量的数据. 然而, 自动生成带程序化特征的 CAD 模型在解决数据稀缺问题的同时, 也面临生成模型与真实工程场景的语义鸿沟. 生成模型的程序化逻辑与真实设计师的设计思维模式存在本质差异, 现有的参数化生成算法往往基于固定的几何规则或随机组合, 难以捕捉人类设计师在工程约束下的渐进式优化过程.

综上所述, 当前研究亟需构建多模态、细粒度标注的标准化评估体系, 并开发基于扩散模型的新型数据生成方法, 以突破真实工程数据获取的

瓶颈限制.

### 3 CAD 表示

在智能 CAD 的发展过程中, 深度学习技术的引入为传统 CAD 的数据表示方法带来了新的视角. 传统的 CAD 数据表示方法, 如以三视图为代表的传统草图、B-Rep 等, 建立在显式几何约束与离散拓扑关系的数学框架之上, 其精确性虽然能够满足基础建模需求, 但是在应对大规模参数化设计、动态迭代设计等场景时, 却暴露出了其局限性: (1) 传统的结构化数据难以有效地捕捉设计意图与功能语义的隐式关联; (2) 基于规则驱动的几何处理算法在面对非结构化设计数据时泛化能力有限; (3) 多层装配关系的建模往往依赖人工定义的约束逻辑, 缺乏适合装配任务自动化的数据表达方式. 与之相比, 深度学习作为一种数据驱动的工具, 能够从海量的设计数据中自动学习到几何形态、拓扑关系、工程语义等的有效特征表示, 为下一步的分析或生成任务提供易于学习的数据表达.

下面将 CAD 表示的现有成果分为 CAD 模型神经表示、B-Rep 表示和装配进行综述, 这不代表 B-Rep 表示没有用到神经表示的方式, 也不是说明装配任务不需要用到 B-Rep 格式的数据, 只是三者处理数据与任务的侧重点各不相同. CAD 表示作为任何智能 CAD 任务的第 1 步, 可以应用于各种下游的分析或生成任务中, 本节介绍的部分工作在创新数据表示方式的同时也兼顾到下游任务的实现, 将注意力主要放在表示方式上, 下游任务则不是重点.

#### 3.1 CAD 模型神经表示

CAD 模型具有高度的顺序化和参数化特性, 设计师常通过一系列精心规划的几何操作构造模型, 这些操作包括但不限于通过曲线草图确定基本轮廓、利用拉伸操作生成实体、通过布尔运算实现多个部件的组合, 以及进行倒角处理以平滑边缘等, 每步操作都由一组离散或连续的参数严格控制, 不仅保证设计过程的精确性, 也使得生成过程充满层次性和逻辑性. 通常, 传统的三维表示方式 (如体素、点云和网格) 依赖固定的离散结构描述空间信息, 显然不符合 CAD 构建流程的非线性和不规则性, 难以直接捕捉到模型的全局几何特征和局部联系. 深度学习的引入为 CAD 模型带来了神经表示的形式, 通过利用神经网络对 CAD 模型进行编码与解码, 能够在在一个紧凑的潜在空间中

高效地捕捉模型的全局几何特性,且又不失对局部细节的精准刻画,简化了传统几何建模的复杂过程,甚至实现对模糊或不完全数据的处理.这种神经表示不仅为模型的生成提供一种全新的方法论,也为后续的优化和几何分析任务奠定了坚实的基础.

CAD 模型神经表示的方法丰富多样,通常需要视后续任务而定,但也不乏典型而通用的方法被后续工作不断沿用与改进,DeepCAD<sup>[5]</sup>和 DeepSDF<sup>[58]</sup>就是其中的代表性的工作.

DeepCAD 是首个基于深度学习的 CAD 模型命令序列生成方法,通过类比 CAD 操作和自然语言,提出一个基于 Transformer 的 CAD 生成网络.该方法将 CAD 模型解读为基于草图的拉伸过程,用草图和拉伸 2 种操作类型表示模型.其中,草图操作包括直线、圆弧和圆 3 种类型,每种操作有相应的参数,三者共同组成草图轮廓;拉伸操作则由草图平面参数、缩放类型、布尔类型和拉伸类型共同定义.图 2 展示了 1 个命令序列表示的示例. DeepCAD 考虑将最常用的 CAD 操作正则化,统一表示为 16 维参数向量,通过量化处理,将连续参数离散化为 256 级整数以保持几何约束关系,这样表示的命令序列有助于神经网络的学习.嵌入的计算被看作 3 个嵌入的和,分别表示命令类型、命令参数和位置编码.基于 Transformer 的自编码器由 4 层编码器和解码器构成,采用前馈策略<sup>[59-60]</sup>预测整个命令序列.该工作还构建了一个大规模的 CAD 命令序列数据集用于网络的训练,生成的命令序列可以直接导入 CAD 工具中进行用户编辑或转换为其他三维表示形式.

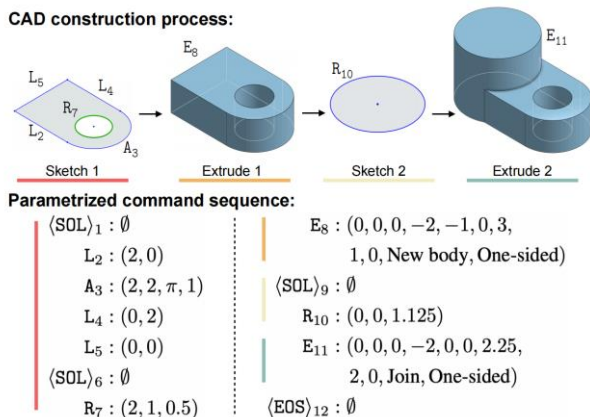


图 2 DeepCAD 命令序列表示<sup>[5]</sup>

该工作开创性地使用命令序列的形式表示 CAD 模型,并提出一个生成命令序列的可行方法;虽然还存在支持的 CAD 操作类型有限、生成序列

的拓扑有效性不足、长序列生成稳定性不足等问题,但草图拉伸的模型构造方式和命令序列的模型显式表示方法都为后续的研究工作提供了宝贵的思路.

DeepSDF 是广泛应用于 CAD 领域的符号距离场(sign distance function, SDF)表示方法,常见于草图的隐式表示. DeepSDF 的 SDF 表示法如图 3 所示,其中,点到表面边界的距离用场的数值表示,场的符号则区分点在形状内部(负)还是外部(正).这种表示形式隐式地编码了形状的边界(零等值面),同时显式地分类了空间. DeepSDF 使用一个潜在编码向量映射到一个由连续 SDF 表示的三维形状,采用联合优化策略,通过自动编码器框架在训练过程中同步学习网络参数和潜在编码向量.潜在编码向量被显式地建模为可学习参数,而不是通过编码器生成,强制网络建立形状特征与潜在空间的直接关联.在推理阶段则固定已经训练好的网络参数,仅优化新形状的潜在编码向量,采用这样的方式使网络参数学习到形状类别的通用先验(如对称性、几何规律),而潜在编码向量捕捉个体形状的细节特征,使得模型可以泛化到新的形状,不需要针对每个形状单独训练.与体素、网格、点云等离散的方法相比,这种连续表示有更好的灵活性和保真度<sup>[61]</sup>. SDF 连续的性质使这种表达方法能够支持任意分辨率的重建,且其几何平滑性与可微性梯度的形状对编辑或生成十分友好,支持端到端的训练,因此在许多重建工作中被广泛使用.

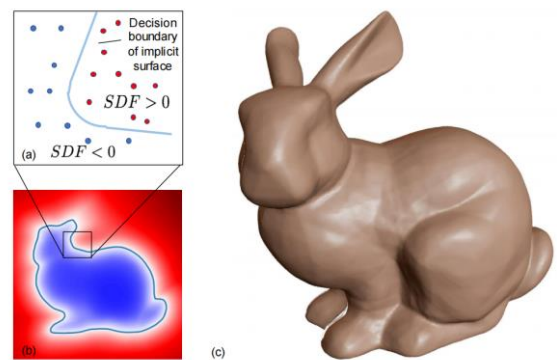


图 3 DeepSDF 的 SDF 表示<sup>[58]</sup>

除了 DeepCAD<sup>[5]</sup>和 DeepSDF<sup>[58]</sup>这 2 篇影响深远的工作外,最近还涌现出大量关于模型神经表示的方法,其中有的是沿用现有方法进行改进,也有的针对新型任务另辟蹊径.下面将该领域中部分前沿工作进行介绍.

(1) SkexGen<sup>[27]</sup>. SkexGen 可以看作是对 DeepCAD<sup>[5]</sup>的改进,它也是草图表示的代表性工作之

一. 在 TurtleGen<sup>[62]</sup> 和 DeepCAD<sup>[5]</sup> 的基础上, SkexGen 将草图和拉伸的命令序列表示定义为原语的层次结构, 由低到高分别为曲线—环—面—草图—拉伸草图, 如图 4 所示. 其中, 曲线是最基础的元素, 支持直线、圆弧、和圆 3 种类型; 环由闭合曲线路径构成; 面由一个外环和可选数量个内环组成; 草图由一个或多个面组成, 支持复杂的二维结构; 拉伸草图是通过拉伸操作将草图升维形成的三维体积. 分层策略能够帮助网络理解模型的语义, 在许多处理相似问题的方法中均有应用<sup>[63]</sup>. 在命令序列的表示上, SkexGen 使用 5 类令牌序列描述建模过程, 分别是指定曲线类型的拓扑令牌、指定曲线几何位置的几何令牌、结束原语令牌、包含拉伸参数和布尔操作类型的拉伸令牌, 以及结束序列令牌, CAD 模型使用这些令牌进行编码. 在网络架构上, 采用基于 Transformer 的双分支自回归架构, 通过解耦的代码本<sup>[64]</sup>控制 CAD

建模序列的生成, 其网络结构分为草图和拉伸 2 个分支: 草图分支中, 使用拓扑编码器和几何编码器分别提取二维草图的拓扑结构(曲线类型)和几何细节(坐标点), 通过向量量化生成独立的拓扑代码本和几何代码本; 拉伸分支则编码拉伸操作的参数生成拉伸代码本. 解码器采用自回归 Transformer, 通过交叉注意力机制融合这 3 个代码本的信息, 逐步生成包含草图绘制(拓扑令牌、几何令牌)和拉伸操作(拉伸令牌)的构造序列. SkexGen 通过代码本的条件采样机制实现设计空间的可控探索, 用户可以通过固定特定代码本(如拓扑代码)保持核心结构不变的同时, 在几何或拉伸维度进行多样化生成.

在草图生成的工作中, 如 SkexGen 这样的令牌序列表示方式十分常见, 原因是其序列化的形式适合于基于 Transformer 网络架构的学习.

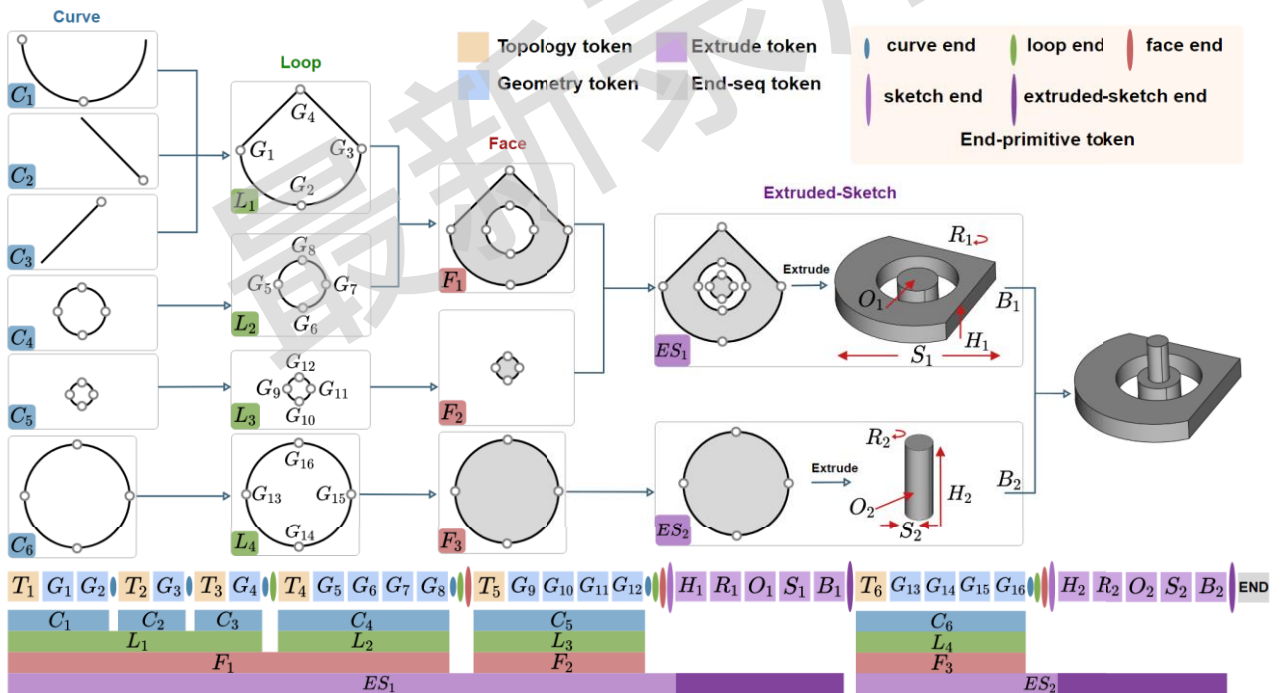


图 4 SkexGen 层次结构<sup>[27]</sup>

(2) HNC<sup>[65]</sup>. 与 SkexGen 相似, HNC 也是基于“草图—拉伸”的数据表示形式将 CAD 模型分层表示. 其中, 模型自下到上被解构为环(Loop)—轮廓(Profile)—实体(Solid)这 3 层树结构, 如图 5 所示. 底层环节通过有序坐标序列表示封闭曲线(如线段、圆弧); 轮廓节点通过二维边界框参数描述多个轮廓的空间排布, 包含一个外环和 0~N 个内环; 而实体节点则通过三维边界框参数记录多个拉伸轮廓的组合关系. 这种层次化结构完整地保

留设计意图的几何与拓扑约束, 为后续神经编码树的学习提供了结构化基础. SkexGen 和 HNC 中, 前者的拓扑、几何和拉伸代码本间的依赖关系通过 Transformer 的注意力机制隐式学习, 但缺乏显式约束, 修改时无法同步跨层级参数; 而后者在解耦建模操作的基础上使用显式树状结构直接绑定层级关系, 对某一层级的修改会带动其他层级的联动调整. 在编码阶段, HNC 中设计了改进的向量量化 VAE(vector quantized VAE, VQ-VAE)<sup>[64,66]</sup>, 提出

带有“掩码跳跃连接”的三级代码本学习机制,分别学习环、轮廓和实体码本;在生成阶段则采用级联式自回归 Transformer 架构.模型编码器首先将部分输入序列转换为潜在嵌入,代码树生成器通过交叉注意力机制预测层次化神经代码,模型生成器则联合代码树与输入嵌入完成最终建模序列生成.代码树采用深度优先遍历表示,通过特殊的分隔符标志层级边界,使得用户可以通过编辑代码节点实现多粒度控制:全局层(实体代码)调整部件布局,中层(轮廓代码)控制截面形态,底层(环代码)微调曲线细节.上述设计使 HNC 支持保持设计意图的编辑操作和智能补全功能,实现可控 CAD 生成.

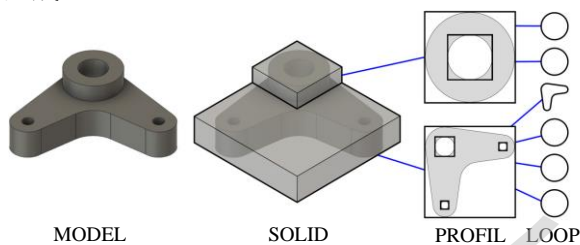


图 5 HNC 模型树结构<sup>[65]</sup>

(3) MultiCAD<sup>[67]</sup>. MultiCAD 是首次针对多模态 CAD 模型表示学习的工作,它将 CAD 模型视为多模态数据,提出一种结合几何信息和构造序列的统一表示方法. CAD 模型的几何形状和构造序列是 2 种互补的信息来源:几何形状(该方法中用点云表示)提供整体的结构和细节,构造序列则反映了设计的逻辑和步骤.基于这一思想,MultiCAD 中提出一种基于对比学习的多模态融合框架,针对点云和构造序列 2 种模态,分别使用 PointNet++<sup>[68]</sup>和 Transformer 编码器进行特征提取.为了克服模态差异,建立跨模态的关联,MultiCAD 中设计一种多模态对比损失,将序列与点云模态的余弦相似度进行几何平均作为跨模态相似度度量,2 种模态的相似度合并为统一指标,不同 CAD 模型的嵌入向量仅在 2 种模态表征均相似时才会彼此靠近.通过对比不同 CAD 模型的相似度,能够对齐跨模态的特征,同时规避了序列与点云模态间的结构差异问题.针对 CAD 数据的特殊性,MultiCAD 预处理数据时对 CAD 模型的几何形状进行随机变换,同时同步地调整构造序列的对应参数,实现跨模态的数据增强,在保持拓扑有效性的前提下扩展了数据多样性;并将深度学习处理多模态数据的方法<sup>[69-70]</sup>引入 CAD 模型的表示学习中,针对 CAD 任务的特殊性进行了有趣的创新.

### 3.2 B-Rep 表示

B-Rep 表示是一种通过描述实体的边界几何和拓扑信息定义三维 CAD 模型的方法,以几何元素和拓扑结构的结合为基础,完整地表示三维物体.其中,几何部分包括顶点、边和面,这些元素通过精确的数学描述(如点坐标、曲线方程和曲面方程)定义物体的形状;拓扑部分描述这些几何元素之间的关系,包括哪些顶点构成边、哪些边定义面,以及面与面如何连接等.

B-Rep 支持高精度的几何表达,能够精确地描述如 NURBS 的复杂曲面,以及多面体、孔洞的拓扑结构,满足工程设计与制造的严苛要求,其显式存储的拓扑关系为模型编辑、碰撞检测等操作提供了结构化基础.此外,B-Rep 与工业标准(如 STEP, IGES)兼容,已经成为 CAD 数据交换的标准;同时,B-Rep 也是一种非常昂贵的数据表达方式,其数据结构复杂度高,包含顶点、边、面、环等多层次拓扑元素,导致存储冗余和计算开销大.B-Rep 具有拓扑关联性,针对 B-Rep 的局部修改可能引发全局拓扑重构,限制了设计的灵活性;离散化特性还使其难以直接融入深度学习框架,传统的神经网络难以处理几何方程加拓扑图那样的异构数据.B-Rep 格式结构化精确性与复杂性并存的特点吸引了许多研究人员的目光,由此也涌现出大量关于学习 B-Rep 表示的工作,包括以 B-Rep 为输入的解析型工作和以 B-Rep 为输出的生成型工作,它们的共同点是针对 B-Rep 设计特定的表示方式,以适配需要处理的任务.由于 B-Rep 格式的拓扑关系表示为点-边-面的 3 层关系,因此许多方法在数据结构的设计上有相当的共同点,即 3 层的层次结构,下面将要介绍的一些工作也印证了这一点.

(1) UV-Net<sup>[28]</sup>. UV-Net 是研究 B-Rep 表示的早期工作之一,将 B-Rep 的几何与拓扑信息解耦表示,模型表示方式如图 6 所示.针对曲面和曲线的几何特征,通过参数域 UV 空间的均匀采样构建规则化网格,每条 B-Rep 的边被表示为一维 UV 网格,记录其三维坐标及单位切向量等特征;每个 B-Rep 的面被表示为二维 UV 网格,捕获了曲面的三维坐标、法向量和裁剪掩码等属性.基于 UV 参数域的网络化表示,UV-Net 将连续的几何信息转化为规则的离散表示,便于神经网络处理.针对拓扑连接关系,通过构建面邻接图进行显式的建模,其中,图的节点表示 B-Rep 中的面,边表示面之间的拓扑连接关系.基于此,UV-Net 采用双路混合架构——通过图像卷积神经网络(convolution neural

network, CNN)提取 UV 网格的局部几何特征, 同时利用层次化 GNN 在拓扑图上进行消息传递, 最

终通过特征融合实现几何与拓扑的联合建模.

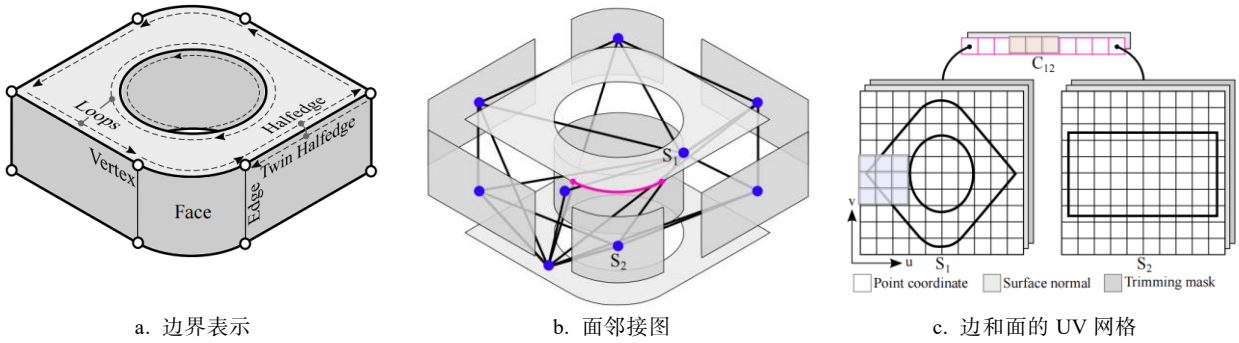


图 6 UV-Net 模型表示方式<sup>[28]</sup>

(2) BRepNet<sup>[44]</sup>. 与 UV-Net 使用的面邻接图不同, BRepNet 使用关系矩阵表示 B-Rep 中的拓扑关系, 以 B-Rep 的拓扑关系为核心设计一种基于共边的卷积机制. 其中, 共边是 B-Rep 中一种有方向的拓扑实体, 表示某个面边界上的一条边及其方向; 每条几何边通常对应 2 个共边, 分别位于 2 个相邻的面上且方向相反, 称为配对共边. BRepNet 共定义 5 种关系矩阵:  $N$  表示当前共边在环中的下一个共边;  $P$  表示当前共边在环中的前一个共边;  $M$  表示与当前共边配对的共边(同一边的反方向共边);  $E$  表示当前共边所属的面;  $E$  表示当前共边所属的边. 通过不同关系矩阵之间的乘积, 沿

着共边的  $N$ ,  $P$  和  $M$  指针构建动态邻域, 将相邻面、边、共边的几何特征进行结构化聚合, 利用拓扑关系显式地编码 B-Rep 的几何特征, 如图 7 所示. 这些特征经过多层感知机(multilayer perceptron, MLP)处理后, 采用池化操作将共边的信息传递至其所属面或边, 解决了多环面的信息流动问题. 利用上述的数据表示, 通过卷积核检测特定操作的特征模式, 该方法可用于处理 B-Rep 的建模操作分割任务. 此外, BRepNet 还构建了一个包含 35000 个标注 B-Rep 模型的数据集, 包含模型每个面的建模操作信息, 用于方法的验证.

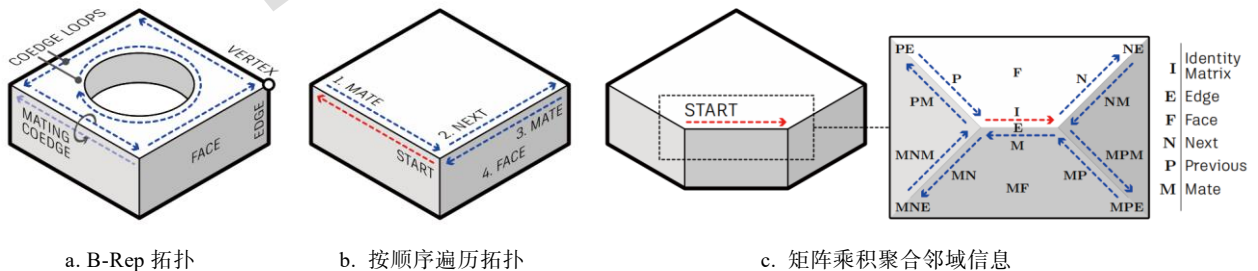


图 7 BRepNet 模型表示方式<sup>[44]</sup>

(3) SolidGen<sup>[40]</sup>. SolidGen 突破了建模操作序列生成的模式, 直接学习并合成 B-Rep 模型. DeepCAD 等方法通过预测草图-拉伸操作序列, 并由几何内核后处理生成 B-Rep, 但存在对建模历史数据依赖性强、难以支持如倒圆角的复杂操作和曲面建模等局限; 而 SolidGen 则巧妙地绕过了这个限制, 设计了索引化 B-Rep, 通过分层引用顶点、边和面构建结构化数值表征. 该方法将 B-Rep 分解为顶点坐标序列, 超边连接索引(2 个点定义直线、3 个点定义圆弧)和面边界索引, 索引逐层引用, 形成

结构化的表示, 既保留了 B-Rep 的几何拓扑关系, 又适应神经网络对结构化数据的处理需求. 在生成阶段, SolidGen 采用分阶段自回归生成架构, 将 B-Rep 生成分解为顶点预测、边预测和面预测 3 个级联模块. 其中, 顶点模块使用 Transformer 解码器量化生成三维坐标序列; 边模块通过指针网络<sup>[71]</sup>动态地引用顶点索引构建曲线基元; 面模块进一步通过跨模态注意力机制聚合边特征生成面拓扑. 这种分层生成策略通过条件概率分解, 建模 B-Rep 的层次化结构.

(4) BrepGen<sup>[41]</sup>. 作为 SolidGen 的后续工作, BrepGen 同样能够生成 B-Rep, 通过设计结构化的隐式几何表示方法, 将 B-Rep 模型编码为具有层次关系的树形结构, 在整体思路与 SolidGen 也是有相似之处的. 具体实现时, BrepGen 对 B-Rep 的处理可分为几何编码和拓扑编码 2 部分.

在几何编码方面, BrepGen 使用边界框实现元素的几何定位, 边界框参数为节点所在三维空间的最小包围盒坐标. 面、边和顶点分别以离散采样点阵表达局部形状, 其中, 面在 UV 参数域<sup>[28]</sup>上均匀采样生成三维点网格, 边沿曲线参数域采样形成一维点序列, 顶点则直接存储三维坐标. 上述几何特征通过 VAEs 压缩为低维潜在向量, 并与全局边界框的位置信息共同构成节点特征, 形成几何的连续隐空间表示. 这样的表示方法将参数化曲面/曲线的几何信息解耦为可学习的全局位置与局部形状特征, 使得模型能够独立地优化位置和形状, 提升对复杂几何的适应能力.

在拓扑编码问题上, 与 SkexGen 的显式表示方式不同, BrepGen 使用节点复制策略隐式地构建连接关系. 共享边或顶点在不同父节点下生成完全相同的子节点, 通过后续节点合并恢复拓扑连接; 同时, 采用随机重复填充策略, 将变长结构(如面的边数量)统一为固定长度树结构, 避免零填充导致的生成偏差. 通过上述方式, 图结构的 B-Rep 拓扑关系被隐式地编码到树结构的重复节点中, 允许后续通过检测重复节点恢复拓扑连接, 确保了拓扑的正确性. 该方法在生成时使用扩散模型, 在去噪过程中通过特征相似性检测重复节点, 动态地合并拓扑关系, 隐式拓扑的表示方法避免显式拓扑预测的误差累积, 适用于复杂连接结构.

扩散模型使用详见文献[41], 此处不再赘述.

### 3.3 装配

装配是 CAD 领域中描述多个零部件之间几何和拓扑关系的重要概念. 装配需要解决 2 个重要问题: 如何选择零件进行装配以及如何组装零件. 前者通常使用相似性分析的方法解决, 智能 CAD 中已有的方法则聚焦于后者<sup>[72]</sup>. 通常, 装配模型通过一组零部件及其约束关系(如位置约束、接触约束和运动约束)定义整个系统的结构和功能, 每个零部件以独立的几何和拓扑表示存在, 而装配模型通过描述零部件之间的连接方式确定它们在三维空间中的装配约束、装配序列等. 多层级装配关系的建模往往依赖人工定义约束逻辑, 设计自动

化程度难以突破瓶颈. 若要使用深度学习的方法将装配的过程自动化, 需要找到合适的表示方式. 传统的显式表示方式(如 B-Rep, CSG)虽然能够完整地保留几何细节, 但是难以捕捉装配体间的装配约束关系. 近年来, 深度学习开始应用于装配领域中, 使用任务相关的表示方式将传统的 CAD 表示(如 B-Rep)转化为适用于神经网络学习的数据对象. 由于装配又与其他 CAD 任务有较大的差别, 表示形式需要适配配对的任务, 具有一定的独特性, 因此单独设置一节介绍关于装配的工作.

(1) AutoMate<sup>[46]</sup>. AutoMate 是首个基于 B-Rep 实现智能化装配的工作. 与大多数关于 B-Rep 的工作相同, B-Rep 数据需要被分层编码, 核心是结构化 B-Rep 图卷积网络(structured BREP graph convolution networks, SB-GCN), 将 B-Rep 建模为包含面、环、边和顶点 4 类实体节点和 3 类层级关系(顶点→边→环→面)的异构图, 并通过元路径(面←环←边→环→面)连接相邻面以缩短图的直径, 如图 8 所示. 为了克服传统 GCN 在异构图上效率低下的问题, SB-GCN 采用分层消息传递机制, 分为 3 个阶段: 首先是自底向上的信息传递, 从顶点到边、边到环、环到面逐层聚合边界信息, 使每个拓扑实体编码其子结构的几何特征; 然后是面层级的元路径卷积, 通过添加面与面之间的元路径连接(如共享边的相邻面)扩大面的局部感受野, 利用多层残差图卷积捕获面之间的几何关系; 最后是自顶向下的信息传播, 将面层级全局信息反向传递至环、边和顶点, 使低维实体能够融合高层语义信息. 这种分层结构在保留异构图信息的同时, 显著地降低了计算复杂度.

在应用层面, 配对被建模为 2 个部件上配对坐标系(mating coordinate frames, MCFs)的约束关系. MCFs 由拓扑实体定义, 其中, 原点通常由面、边或顶点等拓扑实体的几何属性决定, 方向则由拓扑实体的参数化几何特征推导而来. SB-GCN 输出的拓扑实体嵌入与全局零件特征被拼接后输入到分类器, 完成 2 个任务: 一是配对位置预测, 对用户选中的 2 个零件面, 通过对齐坐标系生成相邻的 MCF 候选对, 并基于嵌入相似度进行排序; 二是配对类型分类, 对给定的 MCFs 对预测其约束类型(如固定、旋转、滑动等). 在实际应用中, AutoMate 还实现了一个交互式推荐系统, 能够实时地推荐最有可能的配对位置.

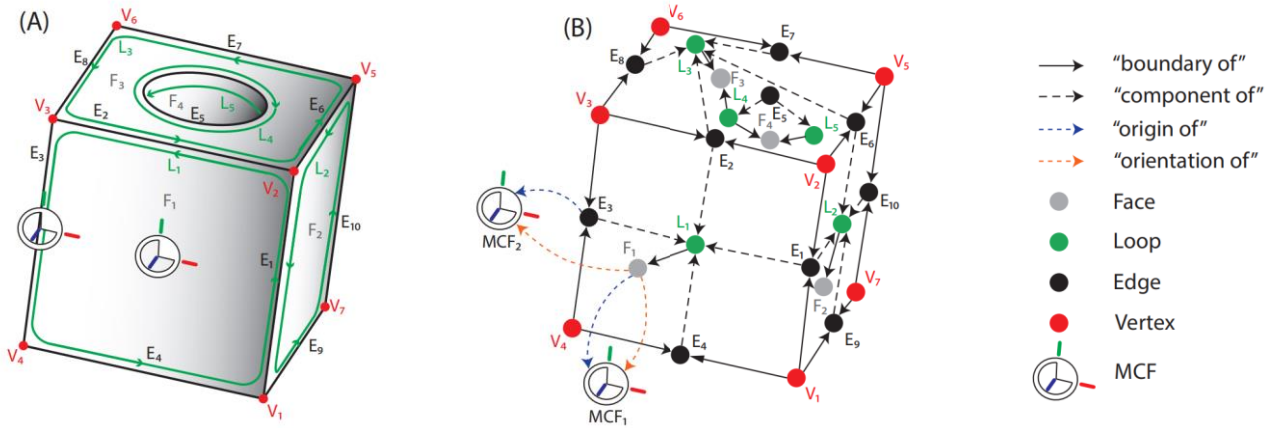


图 8 AutoMate 的模型异构图<sup>[46]</sup>

(2) JoinABLE<sup>[23]</sup>. JoinABLE 是一种面向参数化 CAD 装配任务的弱监督学习方法, 重点解决局部装配关节预测问题, 完全依赖 CAD 输入中的几何和拓扑信息, 无需人工标注或部件类别标签. 该方法将 B-Rep 零件模型表示为图结构, 其中, 节点代表几何特征(面或边), 边表示几何元素的邻接关系, 如图 9 所示; 2 个零件共同组成一个关节, 将装配的过程分成关节轴预测和关节姿态搜索 2 部分. AutoMate 适用于已有明确配对区域且需要用户介入的场景, 更注重提高人机协作效率; 而 JoinABLE 适用于全自动化的局部约束装配任务, 尤其是在缺少明确装配规则的情况下具有优势.

关节轴预测时, 通过双分支图注意力网络提取局部几何特征, 使用全连接方式将 2 个零件图中的所有节点两两相连, 形成一个密集链接矩阵, 将

关节轴预测建模为图链接预测的问题, 后者可看作是一个二分类问题. 针对该问题存在的正样本数量较少的问题, JoinABLE 使用定义等效标签扩展正样本等方法改善问题. 最终通过概率排序确定最优连接对, 实现从数十万候选边中定位有效关节连接, 得到关节的轴.

在关节姿态搜索阶段, 基于上一阶段的关节轴信息在参数空间迭代搜索 3 个参数: 搜索沿关节轴的偏移距离、绕关节轴的旋转角度和轴方向的翻转. 通过调整这些参数, 最小化零件间的重叠体积并最大化接触面积, 实现合理位姿的预测. 该方法首次实现无需人工标注或类别先验的 CAD 关节自动装配, 并构建了一个零件装配的数据集, 使用该数据集在关节轴预测任务中达到 79.53% 的准确率.

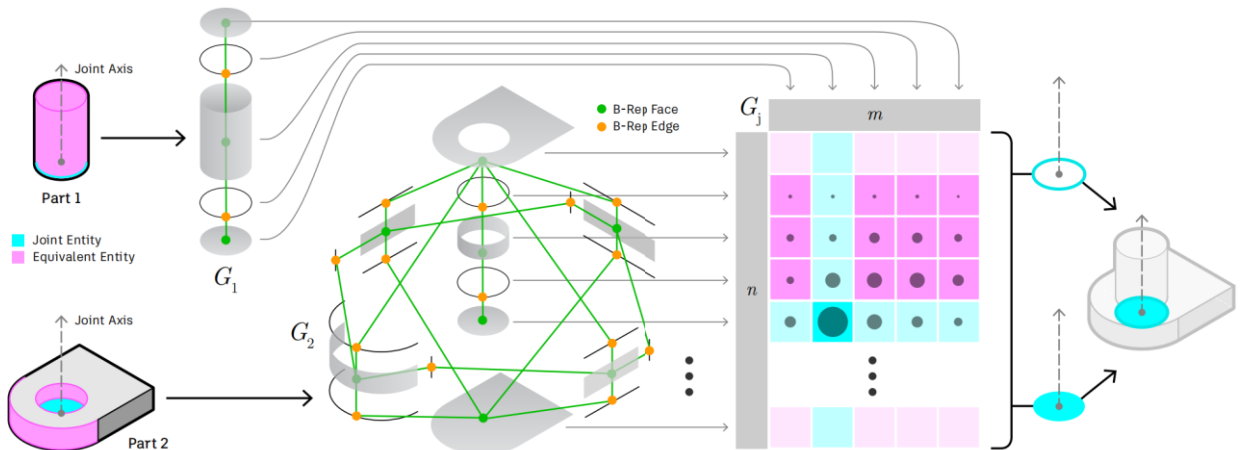


图 9 JoinABLE 模型的图结构<sup>[23]</sup>

## 4 正向工程

在 CAD 领域, 正向工程并不是一个在学术界或工业界常用的词, 为了与常见的逆向工程在设计方向上进行区分, 本文将符合 CAD 常规设计流程的 CAD 任务称为“正向工程”. 正向工程通常需要经历构思→草图→模型的设计流程, 在传统 CAD 领域中, 上述流程需要大量设计人员烦琐且重复的操作, 而智能 CAD 的目标就是利用生成式 AI 的手段将整个设计流程自动化. 本节中, 根据不同的目标对象分成草图生成和基于草图的三维建模 2 部分, 分别对应正向工程的 2 个常见的设计步骤.

### 4.1 草图生成

与位图图像表示的草图相比, 命令序列表示的草图对于智能 CAD 系统显然更具有吸引力. 与位图图像在高维像素空间中隐含的视觉特征相比, 命令序列通过离散化的操作指令和参数化属性, 为神经网络提供了更具语义解释性的学习目标. 在草图生成领域中, 作为其中一种模型表示任务, 大部分的工作都选择将草图信息进行编码表示(或者说设计一种表示草图的语言), 并利用基于 Transformer 的网络架构学习和生成草图, 该流程与 3.1 节讨论的 CAD 表示是一致的. 下面介绍一些有代表性的工作.

(1) SketchGen<sup>[29]</sup>. SketchGen 设计一种形式化语言, 将 CAD 草图编码为结构化符号序列, 每个

草图被分解为原语序列和约束序列, 如图 10 所示, 其中, 原语表示点、直线等几何信息, 约束表示原语之间的几何关系, 通过特定的语法规则组织成层次化符号流  $Q$ . 原语由类型标志符、构造辅助标记和参数组构成, 约束则由类型标志符和指向原语的索引链组成. 通过  $Q^3$  和  $Q^4$  这 2 个辅助序列, 分别记录符号对应的原语/约束类型和参数语义, 形成三维输入特征  $(Q, Q^3, Q^4)$ . 这种表示方法通过语法规则强制结构化, 如标记新元素开始或标记序列结束, 确保生成的符号流严格符合预定义语法.

参考 PolyGen<sup>[73]</sup>的做法, 草图生成过程采用分阶段自回归架构, 将原语生成与约束生成解耦. 原语生成模型基于 Transformer 解码器, 为了处理参数的特异性, 如坐标、半径的连续型参数经过非均匀量化处理, 采用 K-means 聚类将各参数类型分别离散化为 256 个区间值; 然后通过语法树层次序列  $Q^3$ ,  $Q^4$  和位置索引  $Q^1$  强输入编码, 结合 4 类嵌入求和形成特征向量, 自回归预测参数令牌序列; 原语生成时, 通过语法树限制令牌的有效范围, 确保结构合法性. 约束生成模型采用指针网络架构<sup>[71]</sup>, 将上一步生成原语的嵌入和位置编码的令牌序列作为输入的一部分, 结合用于约束的嵌入和位置编码的令牌生成基于原语序列的约束. 生成草图时, 需要使用约束求解器验证是否满足约束, 并对冲突的约束进行修正.

$S = \mathcal{P}, \mathcal{C}$	
$\begin{aligned} \mathcal{P} &= \Lambda, P, \{\Lambda, P\}, \Omega \\ P &= \textit{point} \mid \textit{line} \mid \textit{circle} \mid \textit{arc} \\ \textit{point} &= \tau^{\textit{point}}, \kappa, \textit{location} \\ \textit{line} &= \tau^{\textit{line}}, \kappa, \textit{location}, \textit{direction}, \textit{range} \\ &\vdots \\ \textit{location} &= x, y \\ \textit{direction} &= u, v \\ \textit{range} &= a, b \\ &\vdots \end{aligned}$	$\begin{aligned} \mathcal{C} &= \Lambda, C, \{\Lambda, C\}, \Omega \\ C &= \textit{coincident} \mid \textit{parallel} \mid \textit{equal} \\ &\quad \mid \textit{horizontal} \mid \textit{vertical} \mid \textit{midpoint} \\ &\quad \mid \textit{perpendicular} \mid \textit{tangential} \\ \textit{coincident} &= \nu^{\textit{coincident}}, \textit{ref}, \textit{sub}, \textit{ref}, \textit{sub} \\ \textit{parallel} &= \nu^{\textit{parallel}}, \textit{ref}, \textit{ref} \\ &\vdots \\ \textit{ref} &= \lambda \\ \textit{sub} &= \mu \end{aligned}$

图 10 SketchGen 草图构造语法规则<sup>[29]</sup>

(2) CAD as Language<sup>[53]</sup>. 几何实体和约束是描述一个草图的基本要素, 与 SketchGen 相似, CAD as Language 也需要生成这 2 个要素. 该方法中, 草图由 PB 信息描述, 该数据格式能够移除冗余信息, 缩减数据体积, 同时可以支持任意复杂结构的精确规范定义. 草图也被解析为一系列令牌, 但表示方式与 SketchGen 有差别. 该方法中, 草图被表示

为三元组  $(d_i, c_i, f_i)$  的序列, 如图 11 所示. 其中,  $d_i$  与  $c_i$  分别表示离散值和连续值,  $d_i$  表示枚举类别, 如几何实体类别、约束类别等,  $c_i$  表示对应的参数值,  $d_i$  与  $c_i$  不能同时激活, 两者中必定有一个值为 0;  $f_i$  是一个布尔值, 表示令牌序列的终止. 序列的第 1 个三元组表示指定的对象类型, 后续的

三元组则与上文的三元组语义相关. CAD as Language 使用一个 Transformer 网络同时生成原语信息和约束信息. 生成过程受到一个状态机解释器控制, 能够跟踪当前生成位置在 CAD 数据结构中的逻辑路径(如对象类型选择、几何参数填充或约束条件设置等阶段), 动态地限制模型可预测的标记范围以确保 PB 格式的合法性. 对于条件生成任务, 模型可以接受手绘草图作为输入, 通过视觉 Transformer(vision transformer, ViT)<sup>[74]</sup>提取输入图

像特征, 在自回归生成过程中引入交叉注意力机制, 将像素信息融入草图结构的概率预测. 与 SketchGen 相比, CAD as Language 可以以任意顺序处理令牌序列, 无需先生成原语再生成约束. CAD as Language 构建了一个包含 470 万个预处理 CAD 草图的数据集, 在此基础上进行实验, 验证了该方法在无条件的生成和图像到草图转换任务中的有效性.

	Triplet	Field		Triplet	Field		
Line	1.	(0, 0.0, False)	objects.kind	Point	8.	(0, 0.0, False)	objects.kind
	2.	(0, 0.0, False)	entity.kind		9.	(1, 0.0, False)	entity.kind
	3.	(1, 0.0, False)	line.is_constr		10.	(0, 0.0, False)	point.is_const
	4.	(0, 0.0, False)	line.start.x		11.	(0, 0.0, False)	point.x
	5.	(0, 0.1, False)	line.start.y		12.	(0, 0.1, False)	point.y
	6.	(0, -0.5, False)	line.end.x		13.	(0, 0.0, True)	objects.kind
	7.	(0, 0.2, False)	line.end.y				

图 11 CAD as Language 三元组表示<sup>[53]</sup>

(3) CurveGen-TurtleGen<sup>[62]</sup>. CurveGen-TurtleGen 中包含 2 种草图表示方式和 2 种草图生成模型. CurveGen 与超图表示相对应, TurtleGen 则与海龟绘图表示相对应.

海龟绘图表示把草图绘制看作一系列用笔画图的动作, 将草图表示为命令序列, 语法结构包含闭环命令和绘制命令. 其中, 闭环命令包括一个表示抬笔操作的开始符, 以及一系列表示笔移动动作的绘制命令; 绘制命令分为直线、圆弧和圆 3 种类型, 分别有不同数量的坐标参数, 坐标被量化至 256×256 网格. 闭环是有序的, 按与原点的距离排序绘制. TurtleGen 基于 Transformer 自回归生成命令序列, 迭代地生成完整草图, 最后转换为超图表示, 直接输出几何与拓扑结构.

超图表示包含顶点集和超边集, 顶点集由二维坐标点构成, 超边集通过连接至少 2 个顶点形成几何基元. 基元类型由超边的基数隐式定义: 2 个顶点构成直线, 3 个顶点通过三点定圆生成圆弧, 4 个顶点通过最小二乘法拟合生成圆. 顶点坐标被量化到 256×256 的离散网格, 增强模型对重复坐标模式的学习. CurveGen 模型参考 PolyGen<sup>[73]</sup>, 先生成图顶点; 再根据顶点生成超边, 生成后的超边隐含基元类型信息, 可通过后处理转换为几何基元, 无需依赖约束求解器.

SketchGen, CAD as Language 和 CurveGen-TurtleGen 为同一年出版, 都选择对草图进行编码表示, 只是在草图具体表示方式与生成流程中存在部分差异, 三者各自的优缺点如表 2 所示.

表 2 草图生成方法比较

比较对象	SketchGen <sup>[29]</sup>	CAD as Language <sup>[53]</sup>	CurveGen-TurtleGen <sup>[62]</sup>
草图表示	基于语法树的层次化语言设计	使用三元组定义结构化对象	超图或命令序列
约束处理	显式生成约束, 依赖外部求解器正则化	隐式通过对象序列化确保约束合法性	隐式编码约束
生成流程	分阶段生成原语和约束	端到端生成完整草图序列	分阶段生成或直接生成命令序列
优势	支持复杂约束生成, 适合需要精确约束的应用	灵活性高, 支持条件生成(如图像→草图)	可解释性强, 无需外部求解器
劣势	依赖约束求解器, 生成速度慢	序列化设计复杂, 可解释性低	对复杂约束的显式支持不足

在上述工作的基础上, Vitruvion<sup>[75]</sup>进行了改进. 与之前的方法相比, Vitruvion 与 SketchGen 最相似, 在草图生成流程上需要分阶段生成原语和约束, 并在最后使用几何约束求解器得到结果. 草图的原语和约束信息用三元组形式的令牌列表

示, 三元组由值、ID 和位置标记组成. 其中, 值标记可以表示原语或约束的类型, 也可以表示原语或约束的相关参数值; ID 标记表示令牌的参数类型; 位置标记表示令牌在序列中的位置索引. 原语生成模型使用解码器 Transformer 架构自回归生成

几何原语序列;约束生成模型使用基于编码器-解码器 Transformer 架构,以生成的原语为条件生成约束,并在训练时向原语中注入噪声,模拟实际生成时的误差传播,增强模型对不完美原语输入的鲁棒性;最后使用几何约束求解器调整原语和约束参数.与之前的工作相比,Vitruvion 最大的创新点是在原语生成阶段引入上下文指导原语生成.上下文有 2 种模式:一是草图前缀补全,利用已有的基元序列作为上下文实现自回归续建;二是图像驱动生成,采用 ViT<sup>[74]</sup>编码手绘图谱,通过跨注意力机制将视觉特征注入原语生成过程.为了提升手绘泛化能力,该方法设计了混合噪声模型,包含基元参数的随机仿射变换和渲染过程的高斯过程扰动,能够桥接数字草图与手绘草图的语义差异.

#### 4.2 基于草图的三维建模

随着智能 CAD 技术逐步实现对二维草图的自动理解,从二维草图到三维模型的智能化建模逐渐变成现实.通过分析草图的几何特征、拓扑关系和隐含设计意图,智能 CAD 系统能够生成参数化三维模型,甚至预测用户未明确表达的结构细节.该技术不仅显著地缩短设计周期,还为复杂工程问题的快速迭代提供了可能.然而,如何精准地捕捉设计语义、平衡自动化与用户控制权,以及处理非规范草图,仍是当前研究的挑战.该领域的部分前沿工作如下.

(1) DLAO<sup>[76]</sup>.DLAO 关注单张二维线图自动重建三维 CAD 模型的问题,输入是三维物体在通用视角下的透视投影线描图,包含可见及隐藏的边与顶点.根据针孔相机模型的原理,在已知二维投影坐标的前提下,每个顶点的三维坐标可通过其深度值  $Z$  反推出来,因此问题转为对顶点深度值  $Z$  的几何约束系统求解问题.其总体思路分为 2 个关键环节:首先是几何约束检测.通过神经网络自动地检测三维空间中边与边之间的几何约束关系(如平行、垂直等),构建更可靠的约束方程组;其次是深度初始值预测.利用神经网络预测顶点深度初始值,为非线性优化提供更接近真实解的起点.由于几何约束通常表现为边之间的成对关系,DLAO 将约束检测转化为序列生成任务.针对输入边集  $\xi$ ,给定查询边  $e_{j_1}$ ,模型需要生成与之存在关系的边序列  $g(e_{j_1}) = (e_{j_1}, e_{j_2}, e_{j_3}, \dots, e_{j_k})$ .为此,首先采用基于 Transformer 的指针网络<sup>[71]</sup>提取边的上下文信息,然后利用自注意力机制捕捉全局边

间关系,最后得到候选边的概率分布.为了避免单边预测误差,利用约束的对称性进行过滤,仅当边  $e_i$  出现在  $e_j$  的预测序列且  $e_j$  也出现在  $e_i$  的预测序列时,才保留这对约束.在深度初始值预测方面,DLAO 使用基于 CNN-Transformer 混合架构的深度预测网络,将线图转换为位图并提取视觉特征,模型同时考虑局部顶点坐标和全局拓扑结构,生成初始深度估计.预测的初始深度值直接作为非线性优化器的初始迭代点,降低陷入局部最优的概率,最终经过优化迭代重建出三维模型.

(2) PlankAssembly<sup>[77]</sup>.该方法面向柜类家具设计的三视图模型生成任务.柜类家具具有规则的几何形态、模块化的拼装逻辑、面接触的刚性连接等性质,为数据表示方式提供了许多便利.PlankAssembly 模拟设计师在 CAD 软件中通过组装木板构建柜体的过程,将三维模型表示为可生成的“形状程序”.模型的基本建模单位是轴对齐立方体,具有 6 个自由度,对应 3 个坐标轴的最小和最大边界值.自由度可以通过数值或与其他木板的附着关系动态地指定,这种表示方式不仅模仿设计师在 CAD 软件中逐步组装木板的工作流程,还通过显式的拓扑关系维护模型的编辑性.模型被编码为有向无环图(directed acyclic graph, DAG),其中,节点表示木板的几何面,边表示面之间的附着约束,从而将三维重建转化为结构化序列生成问题.生成模型使用基于 Transformer 的编码器-解码器架构,将三视图中的二维边序列作为输入,输出对应形状程序序列.输入编码采用多层次嵌入策略,融合量化坐标值、视图来源、边类型等特征,由编码器提取全局几何特征;解码器逐步生成由木板几何参数和附着关系构成的结构化程序序列,其中每个步骤通过自注意力机制捕捉上下文依赖,并利用指针网络<sup>[71]</sup>动态地融合几何实体与附着引用 2 种输出模式.在推理阶段,模型通过拓扑排序约束输出顺序,确保父节点优先生成,并应用几何有效性掩码过滤非法附着,最终输出符合 CAD 编辑规范的 DAG.根据 Transformer 的注意力机制对局部误差的容错能力,以及形状程序对几何意图的抽象表达能力,PlankAssembly 方法能够改善三视图重建方法在噪声或缺失输入下的鲁棒性问题.

(3) Sketch2CAD<sup>[26]</sup>.与 DLAO 和 PlankAssembly 相比,Sketch2CAD 更关注系统与用户的建模实时交互.Sketch2CAD 中,建模过程被形式化为一个映射函数,输入为当前三维模型、用户绘制的二维

草图和视角参数, 输出为对应的 CAD 操作及其参数. 考虑到直接解析二维草图的模糊性, 整个处理流程分为 3 个阶段进行. 第 1 阶段是确定操作类型. 分类网络以用户当前绘制的二维草图、深度图和法线图作为输入, 通过 CNN 架构分析其空间特征, 输出拉伸、倒角、添加/减去和扫掠 4 类 CAD 操作的概率分布. 该阶段充分利用三维模型当前状态的几何上下文, 有效地缩小了可能的操作范围; 针对每种操作类型, 分别采用特定操作的 U-Net 进行分割回归, 并行输出基面概率图和曲线分割图 2 个特征图; 分割结果通过几何拟合转化为 CAD 参数, 每种类型的拟合方法各不相同, 如引入 Bézier 曲线拟合倒角轮廓、使用迭代最近点法重建多边形基面等, 用户也可以介入生成过程, 调整生成参数; 最终输出一系列可编辑的 CAD 几何指令和操作参数, 这些参数能够被传统 CAD 软件直接解析并重建出与用户手绘草图意图匹配的参数化三维模型.

为了满足训练任务的需求, Sketch2CAD 使用合成建模序列和渲染草图的方式构建一个笔画草图-命令序列数据集, 数据样本均按预设的序列长度和命令频率随机生成.

(4) Free2CAD<sup>[47]</sup>. Free2CAD 是 Sketch2CAD 的后续工作. Sketch2CAD 的草图绘制是分段式的, 需要多次经历绘制—识别—生成—绘制的流程, 单次识别只能生成一个草图操作; Free2CAD 则通过自动识别笔画分组的方式解决了这个问题. 在 Free2CAD 之前就有关于笔画分组的工作<sup>[78]</sup>, 而 Free2CAD 用于识别笔画分组的方法也并不陌生, 整体流程在前文中早有相似的案例. Free2CAD 基于序列到序列的 Transformer 架构, 首先通过 CNN 自编码器对单个笔画进行编码(256×256 二值图像的潜在空间映射), 并加入位置编码以保留绘制顺序信息; Transformer 编码器通过自注意力机制建立全局笔画关联, 解码器则以自回归的方式逐步生成分组嵌入向量, 允许组间存在时序依赖关系. 为了应对长序列问题, 引入滑动窗口策略, 将长序列分解为局部上下文相关的子序列进行处理, 确保模型在保持全局一致性的同时降低计算复杂度. 操作重建的流程与 Sketch2CAD 相似, 每个笔画分组对应一个独立的 CAD 操作, 结合已重建的上下文进行参数拟合; 分组笔画可能代表一个不完整的轮廓, 系统基于上下文自动补全为完整的参数化形状; 最后, Free2CAD 也使用与 Sketch2CAD 类似的方法构建了一个数据集, 并用于方法的验证.

## 5 逆向工程

正向工程基于明确的设计规范和技术参数, 通过系统化的建模流程实现, 整个过程建立在已知的物理原理与工程知识基础之上. 相对的, 逆向工程以实物样件为研究对象, 通常处理网格、点云、体素等缺少设计信息的数据, 需要反推其设计原理、结构特征和制造逻辑, 恢复出可编辑的参数化数字模型. 正向工程可通过人类操作完成; 逆向工程在人工介入上却面临本质性局限, 设计师在实物逆向过程中难以精确地解析复杂几何特征间的参数关联, 特别是当面对非规则曲面、隐藏结构或精密装配关系时, 人工建模易产生特征丢失与数据偏差. 因此, 智能 CAD 在逆向工程中扮演着十分重要的角色, 利用基于 CNN 与 GNN 的数据解析方法, 比人类更能捕获到实物数据模型的几何特征和拓扑特征, 从而进行更精细的 CAD 重建.

在大部分逆向工程方法中都会使用 CNN 及其变体对数据进行特征提取, 这是由于点云或体素等逆向工程需要处理的数据通常具有空间结构和局部相关性, CNN 的卷积操作能够很好地捕捉这些局部特征, 如邻近点之间的关系或局部形状模式.

按照不同的实现路线, 逆向工程方法可以分为草图拉伸法、CSG 法和分割拟合法 3 类.

草图拉伸和 CSG 都能体现设计师的设计过程. 草图拉伸法通过二维草图轮廓的几何约束推理与三维空间拉伸操作实现模型重构; CSG 法则基于布尔运算原理, 通过基本几何基元(如立方体、圆柱体、球体等)的空间组合构建复杂模型; 分割拟合法通过点云分割、特征提取和曲面拟合的三阶段流程实现模型重建. 表 3 所示为对 3 种实现路线的简要评分, 从数据紧凑性、拓扑重建能力、模型表达能力、计算复杂度和可解释性 5 方面对 3 类方法的平均水平进行评估, 特定的方法可能会对某方面的能力进行针对性的提升.

表 3 逆向工程实现路线评分

维度	草图拉伸	CSG	分割拟合
数据紧凑性	○○○○	○○○○	○○
拓扑重建能力	○○	○○○○	○○○
模型表达能力	○○○	○○	○○○○○
计算复杂度	○○	○○○	○○○○○
可解释性	○○○○	○○○○	○○○

草图拉伸法使用草图的隐式表示和拉伸参数

恢复模型, 表达方式简洁、紧凑, 计算难度也相对较小, 但草图表示的方式也限制了模型的拓扑表达方式; CSG 法有紧凑的模型表示和拓扑完整性, 但受制于体素类型的有限性, 对自由曲面建模存在不足; 分割拟合法有精确还原自由曲面的能力, 但存在计算复杂度高、过度依赖分割精度的技术瓶颈。此外, 草图拉伸法和 CSG 都能体现出设计师的设计过程, 有良好的可解释性; 而分割拟合法则更关注几何的特征, 通常缺少对设计流程的关注, 可解释性不如其他 2 种方法。

### 5.1 草图拉伸

正如“草图拉伸”这一命名, 该方法分为 2 步: 第 1 步对扫描或采集到的实体数据进行处理, 借助特征识别算法推测出关键的二维截面或轮廓草图; 第 2 步基于草图使用拉伸操作生成三维实体模型。

讨论草图拉伸的方法时, DeepCAD<sup>[5]</sup>是一个无法绕开的重要话题。作为开创性的工作, DeepCAD 中提出以草图表示和拉伸操作序列为核心的建模思路, 为后续的研究奠定了基础。此后, 大量的研究基于这一实现路线进行拓展和优化, 提升草图建模的表达能力和生成质量。

草图表示是该方法的重要一环, 3.1 节和 4.1 节讨论 CAD 表示和草图生成时也介绍了许多关于草图表示的方式。而在逆向工程领域, 大部分工作都不约而同地使用到类似 DeepSDF<sup>[58]</sup>的隐式表示方式。SDF 由于表达的连续性, 为深度学习提供了一个可微分的距离度量, 具有优秀的表达能力, 因此

被广泛应用于许多草图拉伸的方法和 CSG 的方法。但是, SDF 也有相应的缺点, 高分辨率的 SDF 需密集的体素化存储使得实时交互受限, 并且表面信息提取存在困难, 如需将 SDF 转换为网格形式, 得到的表面不完全光滑、边缘不够锐利<sup>[51]</sup>。

明确了草图拉伸法的基本思路和数据表达后, 下面是一些关于草图拉伸法的工作。

(1) ExtrudeNet<sup>[30]</sup>。Ren 等<sup>[30]</sup>提出一种基于无监督学习的逆向草图—拉伸建模方法 ExtrudeNet, 从点云中解析出可编辑的 CAD 兼容形状表示, 其流程如图 12 所示。为了实现逆向建模, 该方法中设计了 3 个关键模块: a. rBézierSketch 采用特殊设计的有理三次 Bézier 曲线表示闭合草图, 通过约束控制点极角分布和权重参数, 既能够建模自由曲线又能够精确还原圆形、多边形等基本几何, 同时保证曲线的闭合性和无自交特性; b. Sketch2SDF 中提出参数曲线 SDF 的数值计算方法, 通过对参数域均匀采样并计算最近点距离和法向方向, 将任意参数曲线转换为隐式场表示; c. DiffExtruder 实现可微拉伸操作, 通过可微分算法将二维草图拉伸为三维形状, 先将三维点逆变换投影到草图平面, 定位轮廓最近点, 再结合垂直方向距离与二维轮廓的 SDF, 动态地计算三维形状的 SDF, 最终通过可微 Sigmoid 函数转换为占据概率。最后, 利用 CSG-Stump<sup>[79]</sup>模块进行隐式场的布尔运算组合, 最终以无监督方式优化重建损失和几何约束损失。ExtrudeNet 引入了自由曲线表示草图, 提高了对复杂曲面的建模能力。

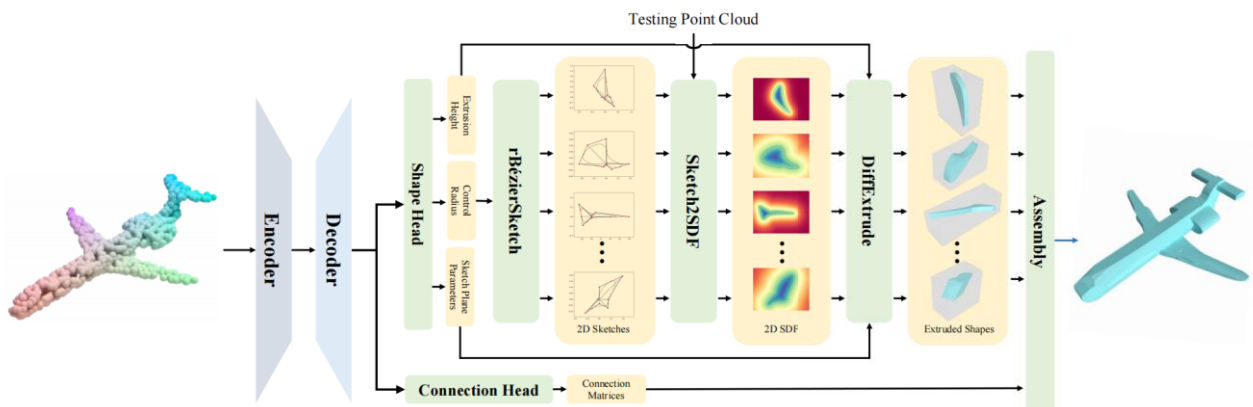


图 12 ExtrudeNet 流程图<sup>[30]</sup>

(2) RepCAD<sup>[80]</sup>。RepCAD 关注从光滑体素模型重建可编辑的棱柱形 CAD 模型, 基本思路是通过分解—重组策略将三维形状表示为可微分的二维轮廓和一维包络函数, 其中, 二维轮廓描述截面形状, 一维包络函数控制拉伸范围; 再结合几何损失

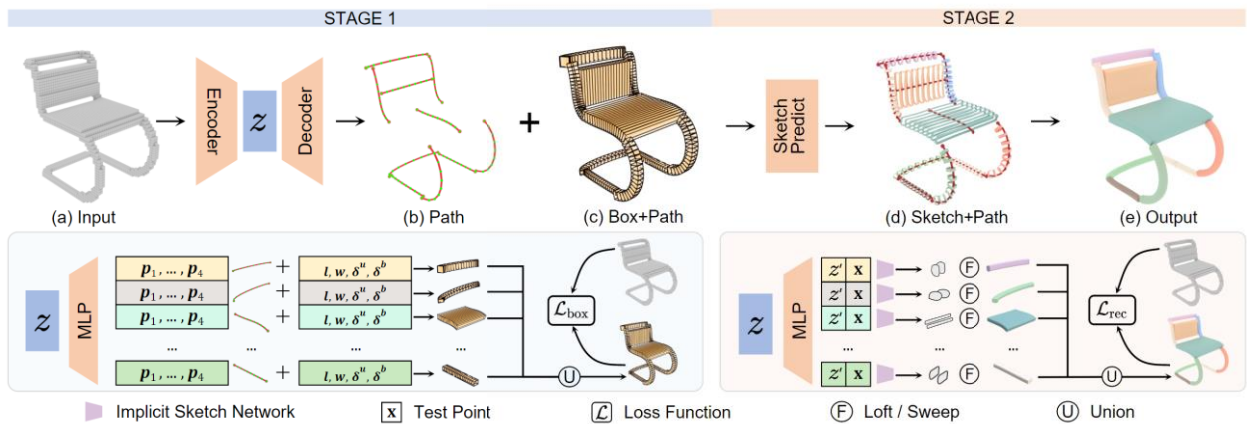
和参数化约束实现高保真重建与可编辑性。RepCAD 首先使用体素编码器<sup>[81]</sup>将体素转换为嵌入向量, 然后通过多个解码模块将其解析为预定义的拉伸操作序列。每个解码模块生成二维轮廓图像和一维包络数组, 其中, 轮廓图像经过下采样后沿

指定轴扩展为三维体素, 包络函数通过最大/最小操作确定拉伸范围, 并通过布尔运算组合各拉伸体素, 最终形成可微分的体素重建结果. 训练过程中采用监督与非监督混合损失, 包括体素交叉熵、轮廓与包络的匹配损失, 确保几何一致性. 在推理阶段, 通过旋转增强与多解码模块评估选择最优重建方案; 然后将生成的二维轮廓转换为符号距离函数, 在预定义的参数化草图库中检索最匹配的约束草图, 并通过优化草图参数实现高精度拟合. 最终, 通过 CAD 内核执行拉伸操作与布尔组合, 生成包含设计意图的可编辑 B-Rep 模型. 该方法背靠 Autodesk 强大的数据能力, 通过检索预定义约束草图, 实现了输出模型的几何合理性与可编辑性.

(3) SECAD-Net<sup>[82]</sup>. SECAD-Net 在 ExtrudeNet 的基础上进行改进. 由于 ExtrudeNet 表达能力不足, 单个几何基元通常比较简单, 输出的结果有大量冗余的几何基元; 而 SECAD-Net 提升了几何基元的表示能力, 能在保证重建精度的同时使用尽量少的几何基元, 使得结果拥有良好的可编辑性. SECAD-Net 首先通过三维卷积编码器提取输入体素的潜在特征; 然后利用全连接层预测多个拉伸框参数, 包括尺寸、中心位置和旋转四元数, 这些参数定义了草图平面的空间定位与轴向方向; 最后采用隐式表示方法, 将三维采样点沿拉伸轴向投影到草图平面, 通过 MLP 生成二维 SDF. 与 ExtrudeNet 相比, SECAD-Net 摒弃了参数化中间表示, 采用端到端的隐式建模直接预测二维隐式场, 草图重建无需显式控制点或几何约束干预, 消除参数化的瓶颈, 提高了几何灵活性. ExtrudeNet 受限于 Bézier 曲线只能生成星形区域, 难以表达复杂的 CAD 特征, SECAD-Net 中该限制也被破除. 三维几何构建阶段, SECAD-Net 设计了可微分拉伸算子, 该算子通过数学公式组合 4 种边界情况, 将二维 SDF 与拉伸高度信息融合, 计算三维圆柱体的隐式距离场; 然后通过 Sigmoid 函数将 SDF 转换为占用概率. 占用组合时使用 Softmax 加权联合

操作融合多个圆柱体, 柱体只通过布尔并集的操作组合起来, 避免了复杂的布尔操作. 实验结果表明, SECAD-Net 的平均输出基元数量是 4.3 个, 远少于 ExtrudeNet 的 14.46 个, 提升了几何表达能力.

(4) SfmCAD<sup>[83]</sup>. 已有的方法通常在学习草图后通过预测一组拉伸操作构建 CAD 模型, 虽然简化了搜索空间, 但同时形状表示能力也相应地削弱了. SfmCAD 考虑拉伸(extrude)、放样(loft)、扫描(sweep)和旋转(revolve)这 4 种操作类型, 加强了方法的形状表示能力. SfmCAD 的实现过程分成了 2 步: 盒子+路径表示, 以及神经草图+路径, 其流程如图 13 所示. 第 1 步中, 参考到 SECAD-Net 中的拉伸框, SfmCAD 利用盒子+路径表示对输入体素进行粗粒度建模, 通过预测 Bézier 曲线的控制点生成沿路径分布的刚性盒子代理; 每个盒子代理的位置由均匀采样的曲线点确定, 其方向通过切向量、法向量和双法向量构建局部坐标系, 并结合可学习的旋转角度参数适应路径的扭曲变化. 该阶段通过联合优化路径控制点与盒子几何参数, 结合平滑性正则化和尺寸约束快速地捕捉形状的整体拓扑结构. 第 2 步聚焦局部几何细节的隐式草图学习. 基于第 1 步得到的路径, 通过 MLP 将全局特征分解为多个局部特征, 结合隐式草图网络预测草图的 SDF. 针对拉伸、放样和扫掠等不同 CAD 操作设计可微分的几何生成方法: 拉伸操作结合草图 SDF 与高度约束生成几何基元; 放样操作通过线性插值上下端草图的隐式场构建过渡几何; 扫掠操作则分解路径为多个线性段, 并计算各段的 SDF 最小值实现形状生成; 旋转操作则被看作一种特殊的扫掠操作, 未对其进行考虑. 为了加速训练, 该方法提出实时采样策略, 在基础草图平面生成采样点后沿路径复制几何信息, 显著地降低计算复杂度. 整个流程通过端到端的损失函数监督, 结合粗粒度盒子代理的快速定位与细粒度隐式场的细节恢复, 最终输出可编辑的草图+路径参数化 CAD 模型.

图 13 SfmCAD 流程图<sup>[83]</sup>

上述方法是典型的草图拉伸方法的逆向工程工作, 还有一些融合了其他类型方法的思路或者有独特的关注点的其他工作, 下面进行简要介绍。

(1) Point2Cyl<sup>[84]</sup>. Point2Cyl 将点云分解为多个拉伸圆柱体, 每个包含二维草图和拉伸参数. Point2Cyl 参考常用于分割拟合法的分类思路, 使用 PointNet++<sup>[68]</sup>主干学习点云的点云的全局特征, 预测每个点所属的拉伸实例及表面属性(基座或桶身); 然后利用闭合解公式计算拉伸轴、中心点、缩放因子和拉伸范围; 最后利用投影和隐式函数重建参数化的可编辑闭合草图。

(2) CAD-SIGNet<sup>[85]</sup>. CAD-SIGNet 的目标是基于点云逆向推断 CAD 的设计历史, 其中, CAD 建模历史解构为草图-拉伸操作的序列化语言表示. CAD-SIGNet 将 CAD 建模历史解构为草图-拉伸操作的序列化语言表示, 使用一种层间跨注意力机制, 在多个 Transformer 块中交替地进行点云嵌入与 CAD 语言嵌入的交互, 使几何特征能够逐层指导设计步骤的生成. 这种自回归的特性使 CAD-SIGNet 实现了用户交互, 模型使用混合采样策略保留概率前 5 名的设计选项, 设计师能够在逆向过程中进行分支选择, 系统将根据当前上下文和点云几何动态生成后续设计建议。

(3) Img2CAD<sup>[51]</sup>. Img2CAD 实现从单幅二维图像直接生成参数化 CAD 建模序列(草图绘制和拉伸操作). 其中引入结构化视觉几何作为中间表示, 利用预训练的 ViT 提取图像特征, 并通过堆叠沙漏网络<sup>[86]</sup>提取量化的线框, 形成初始结构化几何表示; 然后通过嵌入层和 Transformer 编码器对坐标点特征进行编码, 再与图像特征通过交叉注意力融合, 增强几何与语义信息的交互; 最终, 基于 Transformer 的解码器将融合特征转换为 CAD 命令序列, 输出包含命令类型和参数的序列, 可直接由

CAD 内核解析为 B-Rep.

## 5.2 CSG

CSG 是一种经典的几何建模方法, 其核心思想是通过立方体、圆柱体、球体等基本几何基元的布尔运算, 重构目标物体的几何和拓扑结构, 这种层级化的建模方式与人类工程师的设计思维具有天然契合性. 在逆向建模中, CSG 方法通常以点云或体素数据为输入, 通过几何基元识别、参数优化和布尔操作序列推导, 逐步还原原始设计意图。

基于 CSG 的方法研究历史悠久, 与其他逆向工程实现路线相比具有 2 个优势: (1) 其参数化特性使得模型具备高度可编辑性, 设计者可直接调整基元尺寸、位置或布尔运算顺序, 实现快速设计迭代; (2) CSG 模型的数据结构紧凑, 仅需存储基元参数和操作序列, 在存储效率与模型复用性方面表现优异. 然而, 该方法对复杂自由曲面或有机形态的建模能力存在不足, 且过度依赖基元分解, 可能导致特征细节丢失. 目前, 基元识别与布尔操作序列的自动化推理仍是技术难点。

CSGNet<sup>[31]</sup>是 CSG 方法的启发性工作, 虽然处理的对象主要是二维形状, 但是其自动构建 CSG 树的思想依然深刻地影响了后续的工作. CSGNet 主要完成智能形状解析的任务, 能够将二维或三维几何形状自动转化为符合 CSG 建模原理的程序化生成指令. 其将形状解析视为视觉编程任务, 首先使用编码器-解码器架构中的 CNN 提取输入形状的深层特征, 然后门控循环单元按照预定义的 CSG 语法规则递归生成程序指令序列, 如图 14 所示. 程序指令由圆形、立方体等的几何图元, 以及交、并和差 3 种布尔操作构成, 通过上下文无关语法<sup>[87]</sup>保证生成程序的合法性. 该方法突破了传统自底向上解析方法在计算效率上的瓶颈, 通过递归神经网络实现自顶向下的程序生成, 在加快速度

的同时能够产生紧凑且易于理解的建模指令序列。

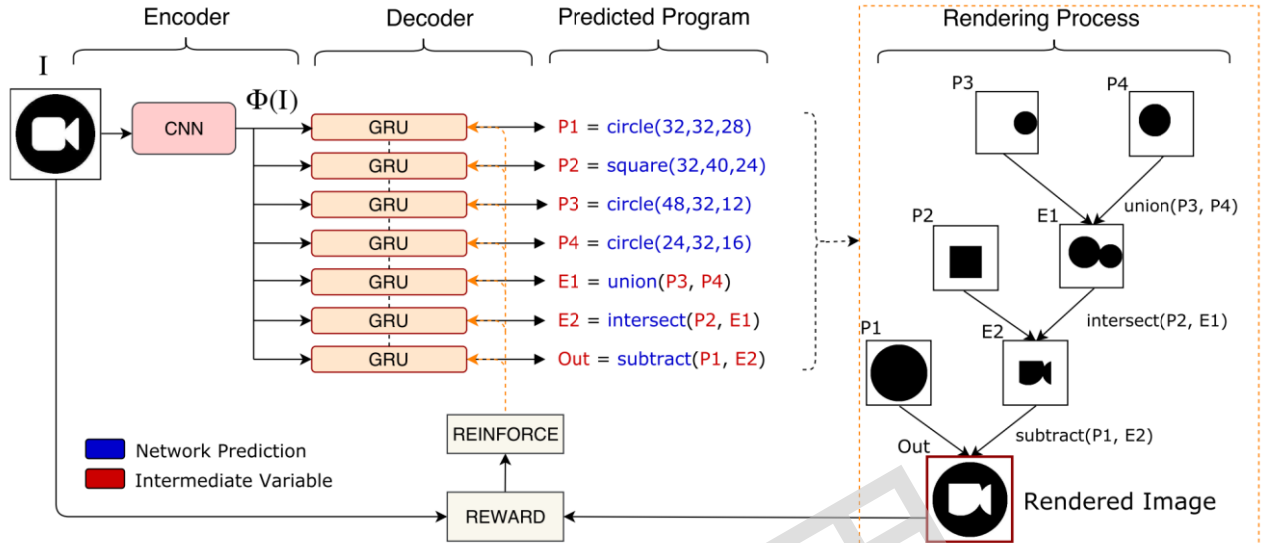


图 14 CSGNet 构造 CSG 树<sup>[31]</sup>

CSGNet 也是一个有监督任务, 训练过程依赖有标注的数据. 其中, 每个形状的解析树由程序自动合成, 标注成本较低. 但将 CSG 任务扩展到三维世界时, 自动生成的方式还有待研究, 为大型三维形状语料库注释解析树需要专业知识和烦琐的注释过程, 标注的成本十分昂贵. 因此, 后续需要找到一种自监督或无监督的方法, 才能将 CSG 的构造方法应用到逆向工程任务中来.

InverseCSG<sup>[88]</sup>结合 RANSAC 和图割的几何检测技术提取基础几何体参数, 利用程序合成器搜索离散布尔结构构造紧凑的 CSG 树; 文献[89]使用遗传算法构造 CSG 树. 上述方法使用基于概率的方法求解问题, 规避了神经网络方法对数据集的需求, 但由于每个形状需要不同的超参数, 通常不能泛化<sup>[90]</sup>.

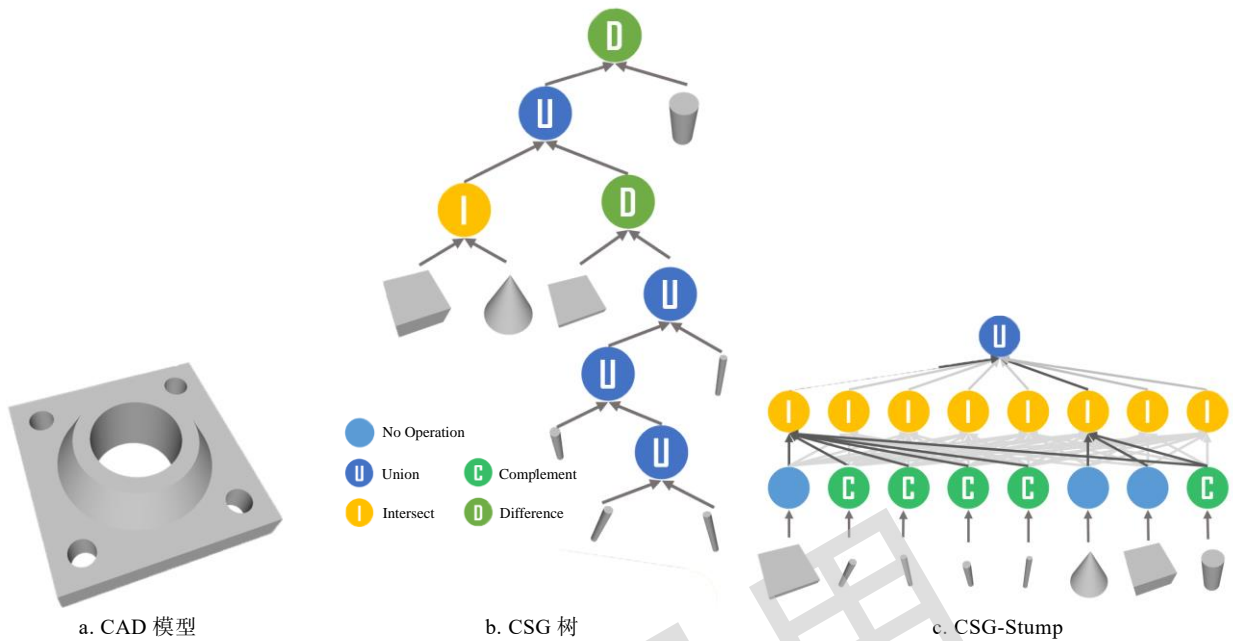
下面介绍将 CSG 方法与神经网络进行深度结合, 探索出易于学习的自监督或无监督的方法.

(1) UCSG-Net<sup>[91]</sup>. UCSG-Net 对 CSGNet 进行改进, 首次实现了完全无监督的 CSG 解析树生成, 无需任何程序标注或中间操作监督. 首先通过编码器将输入对象映射为低维潜在向量; 然后通过基元参数预测网络生成多个几何基元的 SDF 参数, 包括位置、旋转和尺寸信息. 通过可学习的  $\alpha$  参数将 SDF 转换为近似二值化的占用值表示, 将连续距离场转化为离散的内外指示函数, 为布尔操作提供线性计算基础. CSG 层的设计是 UCSG-Net 的

重点, 采用可学习的注意力机制, 结合 Gumbel-Softmax<sup>[92]</sup>重参数化技术, 在多层结构中为每对基元并行地执行并集、交集和差集运算, 通过渐进降低的温度参数使模糊的加权组合收敛为确定性操作路径.

UCSG-Net 无需预定义树结构监督, 仅通过重建损失驱动, 最终将隐式几何表达转化为显式布尔运算树, 既支持复杂非凸形状的分解重构, 又生成了可以直接导入 CAD 软件编辑的符号化建模流程.

(2) CSG-Stump<sup>[79]</sup>. 传统的 CSG 树是深度可变的二叉树, 其拓扑结构随模型复杂度变化, 节点数和分支深度不受限制, 这一特点与神经网络需固定维度的张量是矛盾的, 不适用于深度学习方法中. UCSG-Net 通过多层 CSG 树迭代生成高度可解释的形状, 但由于优化难度大, 只能支持少数层(5 层), 极大地限制了形状的多多样性和表示能力. CSG-Stump 的一大贡献就是提出一种面向深度学习的 3 层结构 CSG 表示方法, 其结构如图 15 所示. 3 层结构分别为底层补集层(C)、中层交集层(I)和顶层并集层(U), 三者分别通过一个连接矩阵进行编码, 表示下层的哪些元素需要执行当前层的操作, 其中, 补集层的操作对象为几何基元. 这种分层结构通过调整连接矩阵的配置, 能够等价表达任意深度的 CSG 树结构, 从而继承了 CSG 的完整表达能力.

图 15 CSG-Stamp 中的 3 层 CSG 表示结构<sup>[79]</sup>

此外, CSG-Stamp 中还设计了 CSG-Stamp Net 架构, 采用 DGCNN 点云编码器提取特征, 通过双分支解码器分别预测几何基元参数和连接矩阵权重; 用 SDF 的方式表示基元, 并转换为占用率; 采用松弛的加权 Softmax 操作替代传统布尔运算, 以解决梯度消失问题. CSG-Stamp Net 通过重建损失和基元损失联合优化, 实现无监督训练. 推理时, 连接矩阵通过阈值二值化生成最终的 CSG-Stamp 结构.

(3) CAPRI-Net<sup>[90]</sup>. CSG 法用于布尔运算的几何基元通常是立方体、圆柱体、球体等简单几何, 而已有方法中多数使用 SDF 的隐式表示方式表达几何基元, 这也许可以用一些更简单、可解释性更好的表示方式去替代. CAPRI-Net 使用一种解释性更强、更紧凑的方式——二次曲面表示几何基元. 首先通过编码器将输入的三维体素形状转换为潜在编码; 然后通过 MLP 预测一组二次曲面基元的参数, 每个基元由方程  $ax^2 + by^2 + cz^2 + dx + ey + fz + g = 0$  定义, 其系数通过 MLP 生成, 这种基元表示方式既增强基元多样性, 又避免了复杂参数化带来的优化难题. 这些基元通过选择矩阵进行交集操作, 形成中间凸形结构; 然后通过并集操作组合为凹形; 最后通过差集操作生成包含孔洞的复杂几何体. 为了优化训练过程, 网络采用多阶段策略: 阶段 0 使用近似函数计算重建损失, 使梯度有效传播; 阶段 1 切换至精确的极小值操作, 引入加权损失函数, 引导网络生成更准确的凸组合结构; 阶段 2 对选择矩阵  $T$  进行二值量化, 固定基

元组合关系形成确定性 CSG 树, 最终得到紧凑且可解释的隐式表达.

(4) D<sup>2</sup>CSG<sup>[93]</sup>. D<sup>2</sup>CSG 是 CAPRI-Net 的后续工作, 从中可以看出许多 CAPRI-Net 的影子. 其核心方法是将形状分解为覆盖形状与残差形状的差集, 其中, 覆盖分支通过基元交集与并集操作构建主体形状, 残差分支则专门构建需减去的复杂补集, 该设计使得布尔操作的顺序不再固定, 差集操作也可以作为 CSG 序列的第 1 步. 每个分支均采用独立的参数化二次曲面集合, 其中, 覆盖分支的二次曲面与 CAPRI-Net 相同, 而残差分支的方程通过改变方程的前 3 个系数符号扩展了形状组合能力, 这种互补基元设计增强了模型对孔洞和高亏格结构的表达能力. 基元参数通过轻量级 MLP 从优化的 256 维特征向量生成, 实现了对目标形状的自适应建模. D<sup>2</sup>CSG 的训练策略也沿用了 CAPRI-Net 的方式, 并在此基础上增加了基于重要性度量的 dropout 策略, 动态地剔除对最终形状影响较小的基元和中间凸形. 该机制能够显著地减少 D<sup>2</sup>CSG 的生成基元数量, 提高表示的紧凑性.

### 5.3 分割拟合

通常, 分割拟合法处理以点云为代表的非结构化扫描数据, 并将其转化为参数化 CAD 模型. 按照处理数据的阶段, 可以分成分割与拟合 2 部分的任务. 分割任务需要将无序点云或三角网格分解为具有明确工程语义的拓扑子域, 如机械零件的平面、圆柱面、自由曲面等特征区域, 其本质是识别数据中的几何不连续边界与功能特征分界,

通过聚类分析或深度学习的方式实现数据驱动下的语义解耦. 拟合任务需要将分割后的局部区域转化为精确的数学表达, 通过参数化曲面或解析几何体重构参数化模型.

草图拉伸法和 CSG 法都属于显式建模, 需预先规划建模逻辑且难以适应复杂曲面的高精度重建; 与这 2 种方法相比, 分割拟合法对数据特征的提取有更高的要求, 通过对数据自适应的特征提取, 可以直接从三维扫描数据中挖掘隐含的几何规律与设计意图, 并且以更精确的曲面表示方式恢复模型. 然而, 这种数据驱动的方法也极度依赖输入数据的质量. 现实世界的数据与合成数据不同, 往往存在各种缺陷<sup>[25]</sup>, 包括缺少部件、出现噪声分辨率不足等, 如何提高方法的鲁棒性是分割拟合法需要面临的一大问题.

PointNet<sup>[94]</sup>是分割拟合法的奠基石, 是一种能

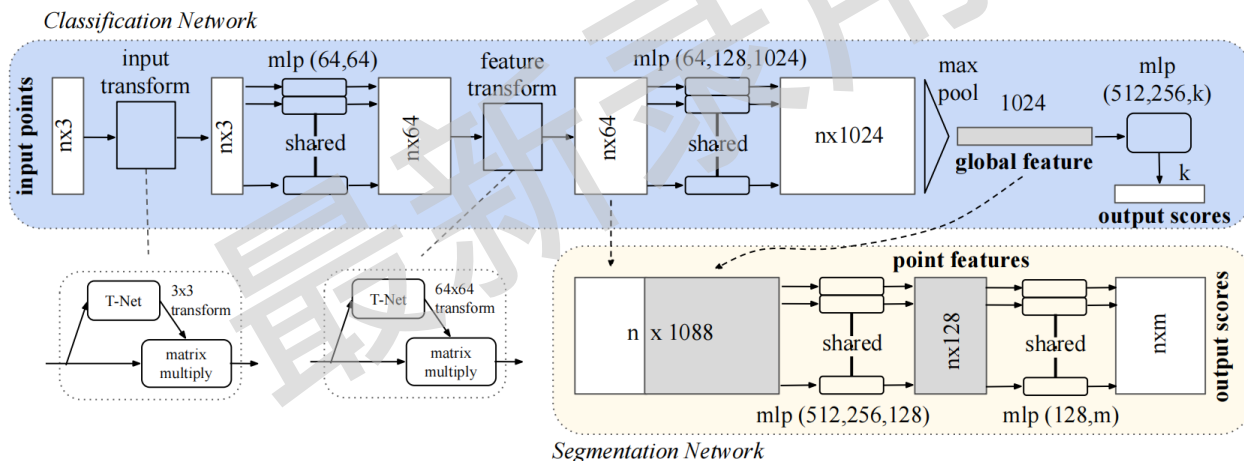


图 16 PointNet 的结构<sup>[94]</sup>

够直接处理点云数据的深度学习框架, 克服了点云输入的无序性和变换不变性问题, 能够高效地学习点云的全局和局部特征, 实现物体分类、语义分割等任务. 图 16 所示为 PointNet 的结构, 其中有 2 个关键点. (1) 为了处理点云的无序性, 首先通过 MLP 独立地提取每个点的特征, 然后通过最大池化将这些特征汇总为全局特征, 保证对输入点顺序的鲁棒性, 使网络对输入点的排列顺序不敏感. (2) 为了实现变换不变性, 引入了两个对齐网络(T-Net): 一个用于对齐  $3 \times 3$  变换矩阵表示的输入点云, 另一个用于对齐  $64 \times 64$  变换矩阵表示的中间的特征空间. 这些对齐网络通过学习点云的几何变换将输入标准化到一个统一的空间, 结合正则化约束确保变换矩阵接近正交, 增强网络对旋转和平移等几何变换的鲁棒性.

PointNet 本身不是用于处理 CAD 任务, 但其直接处理点云数据的能力使得从三维扫描数据中提取物体的几何特征和结构信息成为可能. 通过对点云进行分类和分割, PointNet 能够识别和提取物体的不同部分, 为后续的 CAD 模型重建提供精确的输入; 此外, PointNet 的高效性和对数据丢失的鲁棒性, 使其在处理实际扫描数据时表现出色, 减少数据预处理的复杂度, 提高了逆向工程的效率和精度.

在 PointNet 之后, 一些工作对其进行了改进, 还有许多工作在 PointNet 的基础上构建了基于分割拟合路线的分割拟合方法, 如图 17 所示.

PointNet++<sup>[68]</sup>中引入一个分层神经网络, 递归地将 PointNet 应用于输入点集的嵌套划分中. 利用度量空间距离, 网络能够随着上下文尺度的

增加而学习局部特征, 并使用一个新的集学习层自适应地组合来自多个尺度的特征. SPFN<sup>[32]</sup>使用 PointNet++进行信息提取, 并利用提取的信息进行平面、球体、圆柱体、圆锥的原语拟合, 该工作也是较早的进行原语拟合的工作. CPFN<sup>[95]</sup>是 SPFN 的变体, 使用级联 SPFN 的方式, 每个 SPFN 学习局部信息, 最后进行聚合以处理高分辨率点云. ParSeNet<sup>[45]</sup>也是在 SPFN 的基础上引入了 B 样条曲面的拟合, 同时通过改进拟合网络提高拟合准确率. HPNet<sup>[96]</sup>引入语义信息和局部形状几何的联合监督机制, 填补了已有方法主要使用语义进行监督而忽视结合几何特征的缺点; 同年在 PointNet 的基础上, PrimitiveNet<sup>[97]</sup>结合稀疏卷积方法提取多尺度特征, 处理高分辨率的点云; 之后, ComplexGen<sup>[98]</sup>被提出, 为分割拟合方法提供了一个崭

新的思路,将输入的点云数据直接重建成 B-Rep 表示,重点考虑模型的拓扑学习,在几何方面则使用一个耗时的优化过程进行拟合;当前,SpelsNet<sup>[99]</sup>将已有方法中对几何特征的学习与 ComplexGen 中对拓扑特征的学习相结合,将两者进行联合优化. ComplexGen 与 SpelsNet 的结果均表明,对于拓扑的学习能够使分割拟合方法在有缺陷的数据时展现出良好的鲁棒性;虽然结合了两者的特

点的 SpelsNet 在面的分类分割任务上也显示了较好的结果,但在边的分类分割任务上依然存在欠缺,需要后续工作的进一步改善.此外,SED-Net<sup>[100]</sup>和 Point2CAD<sup>[101]</sup>也在拟合任务上取得进步.在 ParSeNet 和 HPNet 的基础上,SED-Net 添加了对边缘信息的考虑,结合边缘信息进行联合拟合. Point2CAD 则使用一种创新的曲面隐式神经表示,提高了重建精度.

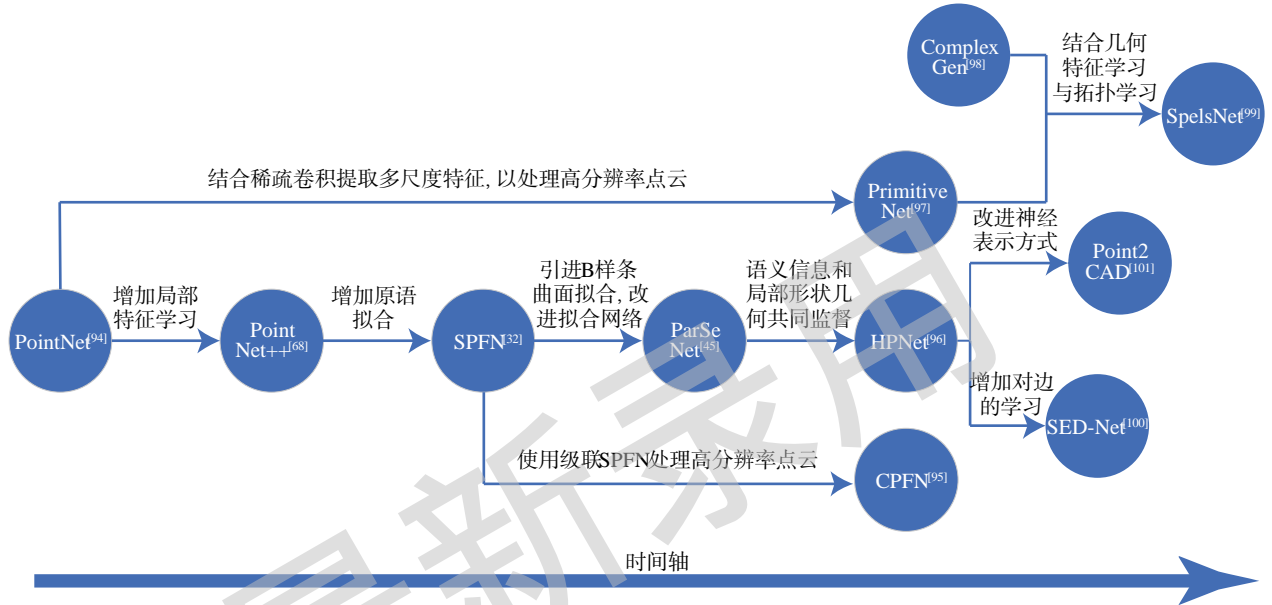


图 17 分割拟合方法总览

## 6 LLM&CAD

LLM(包括 VLM)自问世以来,因其突破性的语言理解与生成能力而受到研究人员的广泛关注,使其能够以类人的方式解析复杂意图、生成创造性内容并实现跨模态知识迁移,彻底改变了传统 AI 在语义交互和逻辑推理领域的局限性. LLM 对 CAD 领域的影响十分深刻,已有的 CAD 方法中,无论是基于传统方法还是智能化方法,都需要以大量的领域专业知识为基础,而 LLM 打破了这个限制,在“知识渊博”的 LLM 的帮助下,即使对于 CAD 领域一无所知的人,也可以通过语言描述构建出结构合理的 CAD 模型数据,在简化交互模式的同时极大地降低了 CAD 领域的使用门槛.

已有很多 LLM 与 CAD 结合的研究,并在上文的任务中取得了初步的成果.下面对已经发表的一些工作进行介绍.

(1) Text2CAD<sup>[24]</sup>. 实现了从自然语言描述到参数化 CAD 模型的端到端生成,核心的生成网络是一个基于 Transformer 的自回归模型而非 LLM,

但这不排除 LLM 在该文献中发挥了重要的作用.该方法基于 LLM 和 VLM 协同的标注策略,成功地构建了包含 17 万模型、66 万文本标注的大规模数据集.

Text2CAD 的数据标注流程分为 2 个阶段: a. 形状描述生成. 首先生成 CAD 模型的多视角图像并交由 VLM 处理,其中使用的是 LLaVA-NeXT 模型<sup>[102]</sup>,能通过对图像的理解生成相应的简化形状文本描述,不涉及具体的尺寸或参数,只关注物体的大致形状,如“一个带有平面顶部和底部的圆柱形物体”或“一个内有孔的圆盘”等.该阶段的目的是提供高层次的、视觉驱动的描述,为后续阶段生成详细设计指令打下基础. b. 多级文本指令生成. 对 DeepCAD 数据集<sup>[5]</sup>中的 CAD 构造序列进行元数据清洗,采用最小元数据生成器规范化参数命名(如将随机键替换为“part\_1”“loop\_1”),并融合第 1 阶段 VLM 提取的形状语义;然后通过 Mixtral-50B 模型<sup>[103]</sup>将结构化参数序列转换为包含几何关系、操作流程的自然语言描述. LLM 将迭代生成 4 种不同层级的指令,分别为抽象层级(L0)、初学者层级(L1)、中级层级(L2)和专家层级(L3),每种层

级的程度和细节不同。

在 Text2CAD 中, LLM 和 VLM 用于处理 CAD 数据注释的任务, 可以体现 LLM 有能力去提取和简化 CAD 模型的信息, 帮助神经网络更好地学习模型。

(2) CAD2PROGRAM<sup>[52]</sup>. CAD2PROGRAM 要解决的问题与 PlankAssembly<sup>[77]</sup>非常相似, 以柜类家具为研究对象, 输入以三视图为代表的二维工程草图, 输出三维参数化模型; 不同的是, CAD2PROGRAM 的实现基于视觉-语言模型 Mini-InternVL-1.5-2B<sup>[104]</sup>, 用更简单、更灵活的方式处理数据。

在输入处理上, CAD2PROGRAM 使用 ViT 架构进行特征提取, 融合几何信息和标注信息(如尺寸标注、功能符号等), 不仅避免复杂的前处理, 还能同时利用几何层的投影信息和注释层的尺寸标注、功能符号等语义线索, 增强了对实际工程图纸复杂性的适应能力。

在输出表示上, CAD2PROGRAM 采用 Python 描述三维参数化模型, 构建文本形式的形状程序。每个原始部件被表示为包含模型 ID、公共参数和模型特定参数的元组, 通过 Python 脚本逐行实例化。与领域特定语言表示方法相比, Python 文本表示避免参数数量随原始模型扩展导致的序列长度爆炸问题, 且 LLM 更擅长处理 Python 编程问题。整个模型使用一种 ViT-MLP-LLM 架构, 其中, ViT 使用 InternViT-300M 处理图像, LLM 使用 InternLM2-1.8B 生成 Python 程序, 两者通过 MLP 投影层将视觉特征与语言模型对齐, 输入提示“Reconstruct cabinet from image:”引导模型生成 Python 脚本程序, 最终脚本经过解析即可重建可编辑的三维参数化 CAD 模型。

(3) CAD-Recode<sup>[33]</sup>. CAD-Recode 是第一个使用 LLM 处理点云逆向工程任务的工作, 实现路线基于上文提到的草图拉伸法, 最大的区别在于使用一个预训练的 LLM 作为解码器, 输出能够重建输入点云几何的可执行 Python 代码。

与 CAD2PROGRAM 类似, CAD-Recode 也需要建立 CAD 操作与编程语言之间的映射关系, 采用 CadQuery 库<sup>[105]</sup>将草图—拉伸序列编码为 Python 代码, 这种表示形式不仅支持基本几何操作, 还能通过函数复用和参数化设计体现复杂建模逻辑。与传统的数值化编码相比, 代码形式天然具备可解释性, 且与 LLM 的预训练知识高度兼容, 能够充分地利用 LLM 的代码生成能力。

CAD-Recode 用轻量级点云编码器与预训练 LLM 解码器组成模型架构, 点云投影模块通过傅里叶位置编码将稀疏点云转化为紧凑的查询向量; 而基于 Qwen2-1.5B 模型<sup>[106]</sup>的解码器则利用其代码生成的先验知识, 将几何特征映射为符合语法和 CAD 规则的 Python 代码。当前, 这种利用 LLM 生成 Python 代码的模式逐渐流行, 充分发挥了 LLM 在序列建模和语义理解方面的优势。

基于 Transformer 的 LLM 虽然有优秀的序列建模能力, 但是在面对点云之类的无序数据时处理能力依然有限, 因此在 CAD-recode 中只作为解码器使用, 需要点云编码器先将初始点云数据转化为适合 LLM 处理的向量序列。这是目前 LLM 技术存在的局限性之一。

(4) Img2CAD<sup>[54]</sup>. 该方法与 Img2CAD<sup>[51]</sup>的名称相同。面对相同的单视图图像逆向生成 CAD 模型的任务时, 两者对 CAD 模型的表示是不约而同的, 均选择了表示草图和拉伸操作的命令序列; 不同点是 Img2CAD<sup>[54]</sup>使用 VLM 提取图像特征。

Img2CAD<sup>[54]</sup>使用 GPT-4V<sup>[107]</sup>对输入图像进行视觉推理, 采用思维链技术<sup>[108]</sup>引导模型生成包含语义部件标签的 CAD 程序框架, 如将椅子分解为椅背、椅腿等部件的离散草图—拉伸操作序列。由于 GPT-4V 在准确地预测 CAD 命令的连续属性方面存在缺陷, 因此在此阶段中其只生成不包含连续参数的离散命令序列, 连续属性的预测则交给其他网络预测。

Img2CAD 中设计了一个名为 TrAssembler 的分层 Transformer 架构, 接受 GPT-4V 预测的离散命令序列为输入, 预测所有 CAD 命令的连续属性。首先使用部件级编码器将每个部件的 CAD 命令序列转换为可学习的连续向量; 然后使用全局编码器整合图像特征与语义标签信息, 利用 GPT-4V 预测的语义标签促进属性空间的信息共享, 形成全局结构感知的部件表征; 最后通过部件级解码器预测每条 CAD 命令的连续几何属性。在此工作中, VLM 虽然不能解决所有问题, 却发挥了不可替代的作用。面对图像处理的任务, VLM 已经能够非常好地完成语义分割的任务, 也许在未来, VLM 可以处理点云领域的语义分割, 取代 PointNet<sup>[94]</sup>的地位。

(5) CAD-MLLM<sup>[55]</sup>. 前文介绍的方法 Text2CAD<sup>[24]</sup>, CAD2PROGRAM<sup>[52]</sup>, CAD-Recode<sup>[33]</sup>和 Img2CAD<sup>[54]</sup>都是利用 LLM 或 VLM 完成特征提取或模型生成的任务; 而 CAD-MLLM 通过设计一个多模态 LLM 框架, 将 CAD 模型的构造命令序

列作为中间表示方式, 利用 LLM 的自回归生成能力, 将文本、图像、点云等多模态输入统一映射到命令序列生成空间。

为了训练多模态模型, CAD-MLLM 还构建了一个包含多视角渲染图像、点云数据、规范化文本描述、构造命令序列的多模态数据集, 针对每种类型的数据分别使用不同的编码器处理。其中, 多视图图像通过视觉编码器 DINO v2<sup>[109-110]</sup>提取特征后再经过交叉注意力层整合, 点云通过点编码器 Michelangelo<sup>[111]</sup>提取特征, 文本描述直接通过分词器<sup>[112]</sup>嵌入, 不同模态特征通过线性投影层对齐到统一语义空间后, 拼接输入至使用 LoRA<sup>[113]</sup>微调的 LLM Vicuna-7B<sup>[114]</sup>中, 以自回归的方式生成 CAD 构造命令序列。

CAD-MLLM 使用多模态的方式处理文本、图像、点云等多种形式的数生成参数化 CAD 模型, 尽管受限于计算资源, 模型的输入受到一定限制(如单次只能接受 2 幅多视角图像作为输入), 但不可否认这是一次大胆的尝试, 实现了一些之前无法做到的事情, 如使用 LLM 处理点云数据, 通过统一框架实现自然语言、视觉与几何数据的协同建模, 将是未来 LLM 与 CAD 结合的新热点。

## 7 未来研究趋势

### 7.1 数据资源

未来, 智能 CAD 仍然需要数据驱动, 数据资源依然是研究的重点, 特别是高质量的带标注 CAD 数据集, 但在现有的成果中这依然是十分稀缺的资源。

在智能 CAD 研究的初期, 数据表示格式以 B-Rep 为主, 之后, 随着研究的深入, 研究人员开始重视命令序列的模型表示方式。B-Rep 形式的数据集缺乏恢复建模操作所需的详细信息, 而命令序列表示能够完整地记录建模操作流程, 更符合神经网络对序列数据的学习机制, 有望成为新一代智能 CAD 系统的标准数据格式。

为了降低用户创建 CAD 模型的门槛, 使非专家也能通过任意多模态条件将他们的想法变为现实, 多模态任务的研究将是未来的重点; 为了应对学习的需求, 还需要构造更多的跨模态数据集。现有的多模态数据集只有 Omni-CAD<sup>[55]</sup>, 未来, 多模态数据集需要在样本数量和模态维度上作进一步的提升; 特别是模态维度上, 要实现 B-Rep、编程语言本、工程图纸等多形态数据的联合编码, 探索

编程语言与几何表示的端到端映射机制, 通过代码编译直接生成 B-Rep 模型, 减少数据中间表示带来的数据损失。

在工业应用层面, 当前的研究尚未覆盖电子 CAD 等专业领域。以 PCB 设计为例, 其布线精度要求达微米级, 需要构建专用的数据集用于训练, 但现有的开源数据集完全空白; 主流 EDA 工具(如 Altium Designer, Cadence Allegro)采用私有数据格式, 不利于数据间的互通, 需要探索数据间的协作互通机制<sup>[115]</sup>, 建立统一的中间表示标准; 为了应对工业上的现实需求, 应当引入 CC3D-ops<sup>[25]</sup>等包含制造缺陷的真实数据集, 通过缺陷样本训练提升模型对工程误差的鲁棒性。

面对数据稀缺的困境, 使用 AI 自动生成 CAD 数据集是一种常用的方式。如 Sketch2CAD<sup>[26]</sup>自动生成笔画和相应的命令序列, Text2CAD<sup>[24]</sup>使用 VLM 和 LLM 生成带有文本描述的数据集。自动生成数据有便捷、低成本等优点, 但也存在着固有的缺陷, 即模型崩溃问题<sup>[116]</sup>。生成方法本身的偏差会在下游任务中持续传播, 导致模型泛化能力下降。生成数据后对数据进行校验, 能够一定程度地缓解这个问题, 但该现象是客观存在的, 难以根治。

最佳的数据集构建方式需要研究社区共同努力, 使用标准化的 API 接口实现数据格式的统一, 如 ABC<sup>[22]</sup>, DeepCAD<sup>[5]</sup>等数据集的构建离不开 Onshape<sup>[57]</sup>的支持, 目前, 命令序列的表示上依然没有统一的方式。研究社区需要构建云端协作的知识共享平台, 通过众包标注、竞赛激励等方式加速数据资源的积累, 破除商业软件的数据壁垒。

### 7.2 处理规模

当前, 智能 CAD 方法普遍存在着处理数据规模小的问题, 而大部分方法在实际使用时通常采用的解决方式是进行采样, 将样本数量进行限制, 这样的方法固然有效, 但也不可避免地损失了部分数据。为了充分地利用输入数据, 需要一种可以提高模型处理数据规模大小的方法。

一种常用的方式是局部学习, 将原始数据进行切片, 对每个局部分别学习, 最后使用各个局部的信息进行聚合学习。SfmCAD<sup>[83]</sup>在模型的路径上将模型分成若干个盒子, 使用盒子近似路径上的曲线, 分成多个局部进行学习, 能更好地捕获局部细节。CPFN<sup>[95]</sup>使用级联 SPFN<sup>[32]</sup>进行学习, 每个 SPFN 学习局部信息, 增强了网络学习高分辨率点云的能力。局部学习的方式在减轻单个网络数据压力的同时, 还为程序的并行运行提供了可能性。

另一种可行的方式是优化数据表达方式, 减少优化参数数量. 在处理高复杂度模型时, 当前的方法需要大量的参数表示以达到精确的拟合, 计算时间呈指数级增加, 特别是那些需要拟合自由曲面的方法. 减少参数数量, 更精简地表示模型是一个有效地减小计算复杂度的方式, 如 CAPRI-Net<sup>[90]</sup>和 D<sup>2</sup>CSG<sup>[93]</sup>针对特定任务精简几何基元的表示方式, 减少需要优化的参数数量, 为预测网络的轻量化提供了数据表示上的支持.

除了减少网络的负担外, 还需要考虑内存与存储的瓶颈问题. CAD 数据通常包含高维几何、拓扑关系和属性信息, 冗余度高, 传统的数据格式(如 STEP, IGES)未针对分布式存储或压缩优化. 上文中介绍的许多关于 CAD 表达的方法中, 大部分表示方式的主要目的是易于学习或生成, 并未提及针对内存进行特别优化. 未来, 处理规模提升之后, 内存是一个不可忽视的问题, 需要使用一些数据压缩的方式. 文献[117-118]对线集进行研究, 可以对未来智能 CAD 发展提供数据压缩的思路.

### 7.3 生成精度

智能 CAD 使用神经网络预测 CAD 任务的结果, 这类方法始终存在准确率方面的问题, 无法像传统的计算方法那样得到完全无误的结果, 当前能做的是尽可能地提升神经网络的生成精度.

加入用户交互是一种可以提高生成精度的取巧方法. CAD-SIGNet<sup>[85]</sup>为用户提供了几种最有可能的生成结果, 在与用户交互的过程中获取用户意图, 并以此指导后续的生成过程, 逐步生成完整的 CAD 数据. 这种方法既能够提高生成精度, 又能够满足用户的个性化生成需求, 具有相当高的实用价值.

在逆向工程中, ComplexGen<sup>[98]</sup>和 SpelsNet<sup>[99]</sup>的案例已经证明, 在恢复模型时增加针对拓扑特征学习的重要性. 学习拓扑的方法能显著地提高缺少部分片段或存在噪声数据的重建精度, 增强模型的鲁棒性, 该特点使得学习拓扑的方法适用于真实世界数据的处理, 具有很大的现实意义. 当前最新的方法依然只是简单地将拓扑特征与几何特征进行联合优化, 更进一步地学习拓扑, 充分利用拓扑带来的结构稳定性和鲁棒性, 相信能够进一步提高智能 CAD 的生成精度.

通常, CAD 中的设计存在重复性和确定性, 基于这样的特点, 可以使用已有数据的先验为数据生成提供初步模板. RepCAD<sup>[80]</sup>就是使用这样的生成方式, 首先在预定义的参数化草图库中检索最

匹配的约束草图, 然后优化草图参数实现高精度拟合, 该方法既简化生成过程, 又提升了生成精度. 然而, 这也是一种昂贵的提升精度方法, 先验库的构建需要强大的数据实力. 未来, 在数据间的共享获得提升之后, 也许可以基于这样的思路高效地提升生成精度.

### 7.4 LLM

已有的使用 LLM(或 VLM)的 CAD 工作已经证明了 LLM 的巨大潜力, 它能在文字序列或者图像中快速地识别 CAD 的设计意图, 并生成指定的离散命令序列或编程语言代码, 同时, 还拥有较高的准确率. 与其他智能生成方法相比, 使用已有的 LLM 进行微调即可执行生成任务, 简化了其他方法中对神经网络的复杂设计工作.

鉴于 LLM 目前对于文本与图像有更强的理解能力, 可以考虑将 CAD 任务中的输入转化为文本或图像形式, 再使用 LLM 对转化后的 CAD 数据进行解析. 如对于点云格式数据的处理中, 可以先使用渲染技术将点云渲染为一幅 CAD 图像, 再效仿 Img2CAD<sup>[54]</sup>的工作使用 LLM 分析图像的信息, 获得准确度较高的提取特征, 做进一步的处理. 但是, 如 Img2CAD<sup>[54]</sup>中提到的, LLM 并不擅长处理连续的参数数据, 需要其他神经网络组件共同工作. 目前, 大部分关于 LLM 的工作都将 LLM 作为数据提取或生成的一个关键组件, 却鲜有使用 LLM 完成完整的数据生成任务的工作.

CAD-MLLM<sup>[55]</sup>为 LLM 的使用提供了新的启发. 多模态 LLM 将多种模态维度对齐, 在 LLM 擅长处理的数据与不擅长处理的数据之间架起桥梁, 使得 LLM 能够接受并理解多种形式的输入. 同时, 多种类型的输入也能为 CAD 任务提供丰富的特征条件, 共同指导数据生成, 提高生成的准确度. 虽然目前的多模态 LLM 工作存在部分输入限制, 但是不可否认它有实现完整 CAD 生成流程的能力, 将是未来 LLM 与 CAD 结合的一个重要趋势.

### 7.5 CAD&CAE

CAD 与 CAE 关注的点并不相同, CAD 的核心是精确地定义和构建产品的几何形状、外观及物理结构, 而 CAE 利用计算机模拟技术分析产品在未来真实使用环境中的性能表现. 在智能 CAD 迎来长足发展的同时, 智能 CAE 领域也取得了很大的进展<sup>[119]</sup>. 在实际的工程实践中, 两者往往是协同工作的. 未来, 智能 CAD 的方法可以考虑将 CAE 需要的一些物理属性加入到学习的对象中. 如在现有神经表示中嵌入材料属性、力学特征等物理语

义,使生成过程同步满足强度、热传导等性能约束;CAE也可以实时地反馈仿真结果,驱动智能CAD进行自动化的设计优化.两者的融合形成设计与性能之间的迭代闭环,相信这正是数字孪生等应用需要的结果.

## 8 结 语

本文系统地回顾了数据驱动智能CAD领域的最新研究进展,首先探讨CAD数据集的构建挑战与优化方向,分析了神经表示、针对B-Rep的表示及装配建模等新型数据表达方式的技术突破;然后介绍常见CAD中设计流程2个方向的一些最新工作,包括正向工程的草图生成与三维建模,以及逆向工程的3种重建路线(草图拉伸法、CSG法和分割拟合法),总结了AI驱动的设计自动化方法;最后结合LLM在自然语言交互与多模态融合中的创新应用,展望智能CAD向语义化、低门槛化发展的趋势.

未来,需要进一步重视数据的作用,突破高质量多模态数据集的构建瓶颈,使用标准化接口与完善开源生态促进CAD领域的数据共享和协同研究,优化神经网络对大规模复杂模型的表示能力,通过拓扑特征学习、用户交互反馈等机制提升生成精度;利用LLM这一工具,LLM与CAD的深度结合将推动跨领域知识迁移与设计意图理解.随着几何推理与逻辑生成技术的协同进化,智能CAD有望在个性化定制、跨学科协同设计与自主创新等场景中释放更大潜力,最终实现从“辅助工具”到“认知伙伴”的跨越.

## 参考文献(References):

- [1] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[C] //Proceedings of the 5th International Conference on Learning Representations. Washington DC: ICLR, 2017
- [2] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C] //Proceedings of the 28th International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2014: 2672-2680
- [3] Kingma D P, Welling M. Auto-encoding variational bayes[OL]. [2025-07-03]. <https://arxiv.org/abs/1312.6114>
- [4] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C] //Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2017: 6000-6010
- [5] Wu R D, Xiao C, Zheng C X. DeepCAD: a deep generative network for computer-aided design models[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2021: 6772-6782
- [6] Brown T B, Mann B, Ryder N, et al. Language models are few-shot learners[C] //Proceedings of the 34th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2020: Article No.159
- [7] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[OL]. [2025-07-03]. [https://cdn.openai.com/research-covers/language-unsupervised/language\\_understanding\\_paper.pdf](https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf)
- [8] Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners[OL]. [2025-07-03]. [https://cdn.openai.com/better-language-models/language\\_models\\_are\\_unsupervised\\_multitask\\_learners.pdf](https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf)
- [9] DeepSeek-AI, Guo D Y, Yang D J, et al. DeepSeek-R1: incentivizing reasoning capability in LLMs via reinforcement learning[OL]. [2025-07-03]. <https://arxiv.org/abs/2501.12948>
- [10] DeepSeek-AI, Liu A X, Feng B, et al. DeepSeek-V3 technical report[OL]. [2025-07-03]. <https://arxiv.org/abs/2412.19437>
- [11] DeepSeek-AI, Bi X, Chen D L, et al. DeepSeek LLM: scaling open-source language models with longtermism[OL]. [2025-07-03]. <https://arxiv.org/abs/2401.02954>
- [12] Zhao Kun, Zhu Longdi, Ding Han. Practical research on optimizing computer aided design with artificial intelligence[J]. Information & Computer, 2025, 37(9): 163-165(in Chinese) (赵坤, 朱隆迪, 丁晗. 人工智能在计算机辅助设计中的应用与实践研究[J]. 信息与电脑, 2025, 37(9): 163-165)
- [13] Zhou Huaifeng. Research on optimisation of clamp assembly based on CAD technology[J]. Agricultural Machinery Using & Maintenance, 2025(2): 87-90(in Chinese) (周华锋. 基于CAD技术的钳工装配优化研究[J]. 农机使用与维修, 2025(2): 87-90)
- [14] Qiu Shufen. Research on AI-based automatic pattern layout technology in garment CAD[J]. West Leather, 2024, 46(14): 52-54(in Chinese) (邱书芬. 基于人工智能的服装CAD自动排版技术研究[J]. 西部皮革, 2024, 46(14): 52-54)
- [15] Li Xue, Zhang Shujian, Zhu Qidong, et al. Research on architectural design patterns based on generative artificial intelligence[J]. Architecture & Culture, 2025(6): 287-290(in Chinese) (李雪, 张枢健, 朱启东, 等. 基于生成式人工智能的建筑方案设计模式研究[J]. 建筑与文化, 2025(6): 287-290)
- [16] Long Hui, Hao Jia, Niu Hongwei, et al. Gesture control technology of CAD model based on situation model[J]. Journal of Mechanical Engineering, 2022, 58(10): 374-382(in Chinese) (龙辉, 郝佳, 牛红伟, 等. 基于情境的CAD模型手势操控技术[J]. 机械工程学报, 2022, 58(10): 374-382)
- [17] Liu Xuehong, Liu Xueyan. Research on the technical characteristics and development of intelligent CAD in the information age[J]. Popular Science & Technology, 2006(1): 30-31(in Chinese) (刘雪红, 刘雪燕. 信息时代智能CAD技术特点及发展研究[J]. 大众科技, 2006(1): 30-31)
- [18] Willis K D D, Pu Y W, Luo J L, et al. Fusion 360 gallery: a dataset and environment for programmatic CAD construction from human design sequences[J]. ACM Transactions on Graphics, 2021, 40(4): Article No.54
- [19] Deng J, Dong W, Socher R, et al. ImageNet: a large-scale hierarchical image database[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2009: 248-255
- [20] Seff A, Ovadia Y, Zhou W D, et al. SketchGraphs: a large-scale dataset for modeling relational geometry in computer-aided design[OL]. [2025-07-03]. <https://arxiv.org/abs/2007.08506>
- [21] Zhou Q N, Jacobson A. Thing10K: a dataset of 10 000 3D-printing models[OL]. [2025-07-03]. <https://arxiv.org/abs/1605.04797>
- [22] Koch S, Matveev A, Jiang Z S, et al. ABC: a big CAD model dataset for geometric deep learning[C] //Proceedings of the

- IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 9601-9611
- [23] Willis K D D, Jayaraman P K, Chu H, *et al.* JoinABLE: learning bottom-up assembly of parametric CAD joints[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 15849-15860
- [24] Khan M S, Sinha S, Sheikh T U, *et al.* Text2CAD: generating sequential CAD models from beginner-to-expert level text prompts[OL]. [2025-07-03]. <https://arxiv.org/abs/2409.17106>
- [25] Dupont E, Cherenkova K, Kacem A, *et al.* CADOps-Net: jointly learning CAD operation types and steps from boundary-representations[C] //Proceedings of the International Conference on 3D Vision. Los Alamitos: IEEE Computer Society Press, 2022: 114-123
- [26] Li C J, Pan H, Bousseau A, *et al.* Sketch2CAD: sequential CAD modeling by sketching in context[J]. ACM Transactions on Graphics, 2020, 39(6): Article No.164
- [27] Xu X, Willis K D D, Lambourne J G, *et al.* SkexGen: autoregressive generation of CAD construction sequences with disentangled codebooks[C] //Proceedings of the 39th International Conference on Machine Learning. New York: PMLR, 2022: 24698-24724
- [28] Jayaraman P K, Sanghi A, Lambourne J G, *et al.* UV-Net: learning from boundary representations[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2021: 11703-11712
- [29] Para W R, Bhat S F, Guerrero P, *et al.* SketchGen: generating constrained CAD sketches[C] //Proceedings of the 35th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2021: Article No.388
- [30] Ren D X, Zheng J M, Cai J F, *et al.* ExtrudeNet: unsupervised inverse sketch-and-extrude for shape parsing[C] //Proceedings of the 17th European Conference on Computer Vision. Heidelberg: Springer, 2022: 482-498
- [31] Sharma G, Goyal R, Liu D F, *et al.* CSGNet: neural shape parser for constructive solid geometry[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 5515-5523
- [32] Li L X, Sung M, Dubrovina A, *et al.* Supervised fitting of geometric primitives to 3D point clouds[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 2652-2660
- [33] Rukhovich D, Dupont E, Mallis D, *et al.* CAD-recode: reverse engineering CAD code from point clouds[OL]. [2025-07-03]. <https://arxiv.org/abs/2412.14042>
- [34] Kim S, Chi H G, Hu X, *et al.* A large-scale annotated mechanical components benchmark for classification and retrieval tasks with deep neural networks[C] //Proceedings of the 16th European Conference on Computer Vision. Heidelberg: Springer, 2020: 175-191
- [35] Chang A X, Funkhouser T, Guibas L, *et al.* ShapeNet: an information-rich 3D model repository[OL]. [2025-07-03]. <https://arxiv.org/abs/1512.03012>
- [36] Wu Z R, Song S R, Khosla A, *et al.* 3D ShapeNets: a deep representation for volumetric shapes[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2015: 1912-1920
- [37] Mo K C, Zhu S L, Chang A X, *et al.* PartNet: a large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 909-918
- [38] Zhang Z B, Jaiswal P, Rai R. FeatureNet: machining feature recognition based on 3D Convolution Neural Network[J]. Computer-Aided Design, 2018, 101: 12-22
- [39] Angrish A, Craver B, Starly B. "FabSearch": a 3D CAD model-based search engine for sourcing manufacturing services[J]. Journal of Computing and Information Science in Engineering, 2019, 19(4): Article No.041006
- [40] Jayaraman P K, Lambourne J G, Desai N, *et al.* SolidGen: an autoregressive model for direct B-rep synthesis[J]. Transactions on Machine Learning Research, 2023, 2023: 1-26
- [41] Xu X, Lambourne J, Jayaraman P, *et al.* BrepGen: a B-rep generative diffusion model with structured latent geometry[J]. ACM Transactions on Graphics, 2024, 43(4): Article No.119
- [42] Cao W J, Robinson T, Hua Y, *et al.* Graph representation of 3D CAD models for machining feature recognition with deep learning[C] //Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. New York: American Society of Mechanical Engineers, 2020: Article No.V11AT11A003
- [43] Colligan A R, Robinson T T, Nolan D C, *et al.* Hierarchical CADNet: learning from B-reps for machining feature recognition[J]. Computer-Aided Design, 2022, 147: Article No.103226
- [44] Lambourne J G, Willis K D D, Jayaraman P K, *et al.* BRepNet: a topological message passing system for solid models[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2021: 12773-12782
- [45] Sharma G, Liu D F, Maji S, *et al.* PARSENET: a parametric surface fitting network for 3D point clouds[C] //Proceedings of the 16th European Conference on Computer Vision. Heidelberg: Springer, 2020: 261-276
- [46] Jones B, Hildreth D, Chen D W, *et al.* AutoMate: a dataset and learning approach for automatic mating of CAD assemblies[J]. ACM Transactions on Graphics, 2021, 40(6): Article No.227
- [47] Li C J, Pan H, Bousseau A, *et al.* Free2CAD: parsing freehand drawings into CAD commands[J]. ACM Transactions on Graphics, 2022, 41(4): Article No.93
- [48] Hähnlein F, Li C J, Mitra N J, *et al.* CAD2Sketch: generating concept sketches from CAD sequences[J]. ACM Transactions on Graphics, 2022, 41(6): Article No.279
- [49] Yuan Z, Shi J Q, Huang Y H. OpenECAD: an efficient visual language model for editable 3D-CAD design[J]. Computers & Graphics, 2024, 124: Article No.104048
- [50] Zhou S D, Tang T Y, Zhou B. CADParser: a learning approach of sequence modeling for B-rep CAD[C] //Proceedings of the 32nd International Joint Conference on Artificial Intelligence. ijcai.org, 2023: 1804-1812
- [51] Chen T R, Yu C N, Hu Y Q, *et al.* Img2CAD: conditioned 3D CAD model generation from single image with structured visual geometry[OL]. [2025-07-03]. <https://arxiv.org/abs/2410.03417>
- [52] Wang X L, Zheng J, Hu Y C, *et al.* From 2D CAD drawings to 3D parametric models: a vision-language approach[C] //Proceedings of the 39th AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2025: 7961-7969
- [53] Ganin Y, Bartunov S, Li Y J, *et al.* Computer-aided design as language[C] //Proceedings of the 35th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2021: Article No.450
- [54] You Y, Uy M A, Han J Q, *et al.* Img2CAD: reverse engineering 3D CAD models from images through VLM-assisted conditional factorization[OL]. [2025-07-03]. <https://arxiv.org/abs/2408.01437>
- [55] Xu J W, Wang C Y, Zhao Z B, *et al.* CAD-MLLM: unifying multimodality-conditioned CAD generation with MLLM[OL]. [2025-07-03]. <https://arxiv.org/abs/2411.04954>
- [56] Li M Y, Fish N, Cheng L L, *et al.* Class-sensitive shape dissimilarity metric[J]. Graphical Models, 2018, 98: 33-42
- [57] PTC. Onshape[EB/OL]. [2025-07-03]. <https://www.onshape.com/en/>
- [58] Park J J, Florence P, Straub J, *et al.* DeepSDF: learning continuous signed distance functions for shape representation[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 165-174

- [59] Carion N, Massa F, Synnaeve G, *et al.* End-to-end object detection with transformers[C] //Proceedings of the 16th European Conference on Computer Vision. Heidelberg: Springer, 2020: 213-229
- [60] Carlier A, Danelljan M, Alahi A, *et al.* DeepSVG: a hierarchical generative network for vector graphics animation[C] //Proceedings of the 34th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2020: Article No.1372
- [61] Dai Tielin, Jin Liuchao, Shang Chen, *et al.* Advances in intelligent design of metamaterials[J]. *Journal of Computer-Aided Design & Computer Graphics*, 2025, 37(1): 1-27(in Chinese) (代铁琳, 金刘超, 尚宸, 等. 超材料的智能设计研究进展[J]. *计算机辅助设计与图形学学报*, 2025, 37(1): 1-27)
- [62] Willis K D D, Jayaraman P K, Lambourne J G, *et al.* Engineering sketch generation for computer-aided design[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2021: 2105-2114
- [63] Niu C J, Li M Y, Xu K, *et al.* RIM-Net: recursive implicit fields for unsupervised learning of hierarchical shape structures[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 11779-11788
- [64] van den Oord A, Vinyals O, Kavukcuoglu K. Neural discrete representation learning[C] //Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2017: 6309-6318
- [65] Xu X, Jayaraman P K, Lambourne J G, *et al.* Hierarchical neural coding for controllable CAD model generation[C] //Proceedings of the 40th International Conference on Machine Learning. New York: PMLR, 2023: 38443-38461
- [66] Razavi A, van den Oord A, Vinyals O. Generating diverse high-fidelity images with VQ-VAE-2[C] //Proceedings of the 33rd International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2019: Article No.1331
- [67] Ma W J, Xu M Y, Li X Y, *et al.* MultiCAD: contrastive representation learning for multi-modal 3D computer-aided design models[C] //Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. New York: ACM Press, 2023: 1766-1776
- [68] Qi C R, Yi L, Su H, *et al.* PointNet++: deep hierarchical feature learning on point sets in a metric space[C] //Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2017: 5105-5114
- [69] Liu Z D, Xing Z, Zhou X D, *et al.* 3D-augmented contrastive knowledge distillation for image-based object pose estimation[C] //Proceedings of the International Conference on Multimedia Retrieval. New York: ACM Press, 2022: 508-517
- [70] Afham M, Dissanayake I, Dissanayake D, *et al.* CrossPoint: self-supervised cross-modal contrastive learning for 3D point cloud understanding[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 9902-9912
- [71] Vinyals O, Fortunato M, Jaitly N. Pointer networks[C] //Proceedings of the 29th International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2015: 2692-2700
- [72] Heidari N, Iosifidis A. Geometric deep learning for computer-aided design: a survey[J]. *IEEE Access*, 2025, 13: 119305-119334
- [73] Nash C, Ganin Y, Eslami S M A, *et al.* PolyGen: an autoregressive generative model of 3D meshes[C] //Proceedings of the 37th International Conference on Machine Learning. New York: PMLR, 2020: 7220-7229
- [74] Dosovitskiy A, Beyer L, Kolesnikov A, *et al.* An image is worth 16×16 words: transformers for image recognition at scale[C] //Proceedings of the 9th International Conference on Learning Representations. Washington DC: ICLR, 2021: 1-21
- [75] Seff A, Zhou W D, Richardson N, *et al.* Vitruvion: a generative model of parametric CAD sketches[C] //Proceedings of the 10th International Conference on Learning Representations. Washington DC: ICLR, 2022: 1-18
- [76] Zheng J, Zhu Y F, Wang K H, *et al.* Deep learning assisted optimization for 3D reconstruction from single 2D line drawings[OL]. [2025-07-03]. <https://arxiv.org/abs/2209.02692>
- [77] Hu W T, Zheng J, Zhang Z X, *et al.* PlankAssembly: robust 3D reconstruction from three orthographic views with learnt shape programs[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2023: 18495-18505
- [78] Yang L M, Zhuang J J, Fu H B, *et al.* SketchGNN: semantic sketch segmentation with graph neural networks[J]. *ACM Transactions on Graphics*, 2021, 40(3): Article No.28
- [79] Ren D X, Zheng J M, Cai J F, *et al.* CSG-stump: a learning friendly CSG-like representation for interpretable shape parsing[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2021: 12478-12487
- [80] Lambourne J G, Willis K, Jayaraman P K, *et al.* Reconstructing editable prismatic CAD from rounded voxel models[C] //Proceedings of the SIGGRAPH Asia Conference Papers. New York: ACM Press, 2022: Article No.53
- [81] Mescheder L, Oechsle M, Niemeyer M, *et al.* Occupancy networks: learning 3D reconstruction in function space[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 4460-4470
- [82] Li P, Guo J W, Zhang X P, *et al.* SECAD-Net: self-supervised CAD reconstruction by learning sketch-extrude operations[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2023: 16816-16826
- [83] Li P, Guo J W, Li H B, *et al.* SfmCAD: unsupervised CAD reconstruction by learning sketch-based feature modeling operations[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2024: 4671-4680
- [84] Uy M A, Chang Y Y, Sung M, *et al.* Point2Cyl: reverse engineering 3D objects from point clouds to extrusion cylinders[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 11850-11860
- [85] Khan M S, Dupont E, Ali S A, *et al.* CAD-SIGNet: CAD language inference from point clouds using layer-wise sketch instance guided attention[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2024: 4713-4722
- [86] Newell A, Yang K Y, Deng J. Stacked hourglass networks for human pose estimation[C] //Proceedings of the 14th European Conference on Computer Vision. Heidelberg: Springer, 2016: 483-499
- [87] Hopcroft J E, Motwani R, Ullman J D. Introduction to automata theory, languages, and computation, 2nd edition[J]. *ACM SIGACT News*, 2001, 32(1): 60-65
- [88] Du T, Inala J P, Pu Y W, *et al.* InverseCSG: automatic conversion of 3D models to CSG trees[J]. *ACM Transactions on Graphics*, 2018, 37(6): Article No.213
- [89] Friedrich M, Fayolle P A, Gabor T, *et al.* Optimizing evolutionary CSG tree extraction[C] //Proceedings of the Genetic and Evolutionary Computation Conference. New York: ACM Press, 2019: 1183-1191
- [90] Yu F G, Chen Z Q, Li M Y, *et al.* CAPRI-Net: learning compact CAD shapes with adaptive primitive assembly[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 11768-11778
- [91] Kania K, Zięba M, Kajdanowicz T. UCSG-NET-unsupervised discovering of constructive solid geometry tree[C] //Proceedings of the 34th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2020:

- Article No.736
- [92] Jang E, Gu S X, Poole B. Categorical reparameterization with gumbel-softmax[C] //Proceedings of the 5th International Conference on Learning Representations. Washington DC: ICLR, 2017: 1-12
- [93] Yu F G, Chen Q M, Tanveer M, *et al.* D<sup>2</sup>CSG: unsupervised learning of compact CSG trees with dual complements and dropouts[C] //Proceedings of the 37th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2023: Article No.989
- [94] Qi C R, Su H, Mo K C, *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. Los Alamitos: IEEE Computer Society Press, 2017: 652-660
- [95] Lê E T, Sung M, Ceylan D, *et al.* CPFN: cascaded primitive fitting networks for high-resolution point clouds[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2021: 7457-7466
- [96] Yan S M, Yang Z P, Ma C Y, *et al.* HPNet: deep primitive segmentation using hybrid representations[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2021: 2753-2762
- [97] Huang J W, Zhang Y F, Sun M W. PrimitiveNet: primitive instance segmentation with local primitive embedding under adversarial metric[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2021: 15343-15353
- [98] Guo H X, Liu S L, Pan H, *et al.* ComplexGen: CAD reconstruction by B-rep chain complex generation[J]. ACM Transactions on Graphics, 2022, 41(4): Article No.129
- [99] Cherenkova K, Dupont E, Kacem A, *et al.* SpelsNet: surface primitive elements segmentation by B-rep graph structure supervision[C] //Proceedings of the 38th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2024: Article No.39
- [100] Li Y Q, Liu S, Yang X R, *et al.* Surface and edge detection for primitive fitting of point clouds[C] //Proceedings of the ACM SIGGRAPH Conference Proceedings. New York: ACM Press, 2023: Article No.44
- [101] Liu Y J, Obukhov A, Wegner J D, *et al.* Point2CAD: reverse engineering CAD models from 3D point clouds[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2024: 3763-3772
- [102] Li F, Zhang R R, Zhang H, *et al.* LLaVA-NeXT-interleave: tackling multi-image, video, and 3D in large multimodal models[OL]. [2025-07-03]. <https://arxiv.org/abs/2407.07895>
- [103] Jiang F. Identifying and mitigating vulnerabilities in llm-integrated applications[D]. Washington: University of Washington, 2024
- [104] Chen Z, Wu J N, Wang W H, *et al.* Intern VL: scaling up vision foundation models and aligning for generic visual-linguistic tasks[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2024: 24185-24198
- [105] CADQuery Developers. CADQuery[EB/OL]. [2025-07-03]. <https://cadquery.readthedocs.io/>
- [106] Yang A, Yang B S, Zhang B C, *et al.* Qwen2.5 technical report[OL]. [2025-07-03]. <https://arxiv.org/abs/2412.15115>
- [107] OpenAI, Achiam J, Adler S, *et al.* GPT-4 technical report[OL]. [2025-07-03]. <https://arxiv.org/abs/2303.08774>
- [108] Wei J, Wang X Z, Schuurmans D, *et al.* Chain-of-thought prompting elicits reasoning in large language models[C] //Proceedings of the 36th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2022: Article No.1800
- [109] Oquab M, Darcet T, Moutakanni T, *et al.* DINOv2: learning robust visual features without supervision[J]. Transactions on Machine Learning Research, 2024, 2024: 1-32
- [110] Darcet T, Oquab M, Mairal J, *et al.* Vision Transformers need registers[C] //Proceedings of the 12th International Conference on Learning Representations. Washington DC: ICLR, 2024: 2632-2652
- [111] Zhao Z B, Liu W, Chen X, *et al.* Michelangelo: conditional 3D shape generation based on shape-image-text aligned latent representation[C] //Proceedings of the 37th International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc, 2023: Article No.3236
- [112] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units[C] //Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: Association for Computational Linguistics, 2016: 1715-1725
- [113] Hu E J, Shen Y L, Wallis P, *et al.* LoRA: low-rank adaptation of large language models[C] //Proceedings of the 10th International Conference on Learning Representations. Washington DC: ICLR, 2022: 1-13
- [114] The Vicuna Team. Vicuna: an open-source chatbot impressing GPT-4 with 90%\* ChatGPT quality[EB/OL]. [2025-07-03]. <https://vicuna.lmsys.org>
- [115] Jones B, Noeckel J, Kodnongbua M, *et al.* B-rep matching for collaborating across CAD systems[J]. ACM Transactions on Graphics, 2023, 42(4): Article No.104
- [116] Shumailov I, Shumaylov Z, Zhao Y R, *et al.* AI models collapse when trained on recursively generated data[J]. Nature, 2024, 631(8022): 755-759
- [117] Kern M, Neuhauser C, Maack T, *et al.* A comparison of rendering techniques for 3D line sets with transparency[J]. IEEE Transactions on Visualization and Computer Graphics, 2021, 27(8): 3361-3376
- [118] Schertzer J, Mercier C, Rousseau S, *et al.* Fiblets for real-time rendering of massive brain tractograms[J]. Computer Graphics Forum, 2022, 41(2): 447-460
- [119] Zhao X W, Tong X M, Ning F W, *et al.* Review of empowering computer-aided engineering with artificial intelligence[J]. Advances in Manufacturing, 2025: 1-41