

DOI:10.11918/202401065

融合图神经网络与概率编码的加工特征识别

赵显文¹, 莫轩东¹, 夏铭远¹, 胡小锋^{1,2}

(1. 上海交通大学 机械与动力工程学院, 上海 200240;
2. 上海市网络化制造与企业信息化重点实验室(上海交通大学), 上海 200240)

摘 要: 为解决多加工特征交叉下的特征定位问题, 提高复杂零件加工特征识别性能, 提出实例分割框架下的加工特征识别方法 Brep3pNet。首先, 基于三维模型的边界表示, 提取面点云、面邻接图等几何与拓扑数据, 构建三维模型的图表示, 利用点云学习网络以及图神经网络学习三维模型面级嵌入表示。其次, 提出概率位置编码方法, 引入位置先验信息将三维模型各面编码为与空间位置相关的三元高斯分布, 基于 Bhattacharyya 核度量面间相似性, 以实现加工特征的面级定位, 生成候选实例。最后, 设计得分网络用于预测实例生成质量, 以此指导实例间的非极大抑制, 去除冗余特征实例, 获得最终加工特征识别结果。本研究在 MFCAD、MFCAD++、MFIInstSeg 和合成的回转类零件数据集等 4 个多特征数据集上对所提方法进行评估。研究表明: Brep3pNet 相较于其他先进方法具有更好的特征定位能力, 可以通过轻量的模型参数实现最优的特征识别准确率, 展现了所提方法在相交特征识别上的应用潜力。

关键词: 加工特征识别; 实例分割; 点云; 图神经网络; 概率编码

中图分类号: TH166 **文献标志码:** A **文章编号:** 0367-6234(2025)04-0116-15

Machining feature recognition based on integration of graph neural network and probabilistic embedding

ZHAO Xianwen¹, MO Xuandong¹, XIA Mingyuan¹, HU Xiaofeng^{1,2}

(1. School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; 2. Shanghai Key Laboratory of Advanced Manufacturing Environment (Shanghai Jiao Tong University), Shanghai 200240, China)

Abstract: To solve the problem of feature localization with multi-feature intersection and improve the performance of machining feature recognition for complex parts, this paper proposes Brep3pNet, a machining feature recognition method under the framework of instance segmentation. Firstly, based on the boundary representation (B-rep) of 3D models, Brep3pNet extracts geometric and topological data such as face point clouds and face adjacency graphs to construct a graph representation of the 3D model. We utilize point cloud learning networks and graph neural networks to learn surface-level embedding representations of the 3D model. Secondly, a probabilistic positional embedding method is proposed, which introduces spatial position prior information to encode the face into ternary Gaussian distribution, and measures the similarity among those face embeddings by Bhattacharyya kernel for the purpose of locating machining features and generating candidate machining feature instances. Finally, a score network is designed to predict the quality of the instance generated, so as to guide the non-maximum suppression between instances to remove redundant feature instances, thereby obtaining the final machining features. Brep3pNet is evaluated on four multi-feature datasets, including MFCAD, MFCAD++, MFIInstSeg and a synthetic dataset of rotary parts. The research results indicate that Brep3pNet outperforms other state-of-the-art methods on feature localization accuracy, and can achieve optimal feature recognition accuracy with lightweight model parameters, demonstrating its potential application in intersecting features recognition.

Keywords: machining feature recognition; instance segmentation; point cloud; graph neural network; probabilistic embedding

加工特征识别技术将存储于 CAD 系统的点、线、面等低级几何与拓扑信息转化到孔、槽等具有明确制造语义的制造特征, 被认为是实现自动工艺规划以及 CAD、CAPP、CAM 系统集成的关键^[1-2]。在

收稿日期: 2024-01-22; 录用日期: 2024-03-28; 网络首发日期: 2025-01-15

网络首发地址: <https://link.cnki.net/urlid/23.1235.T.20250115.1247.002>

基金项目: 国家自然科学基金(51975373)

作者简介: 赵显文(1999—), 男, 硕士研究生; 胡小锋(1977—), 男, 研究员, 博士生导师

通信作者: 胡小锋, wshxf@sjtu.edu.cn

40 年的研究中,研究者们通过归纳各类加工特征的存在模式,提出了多种基于规则的特征识别方法,包含基于图^[3]、基于体分解^[4]以及基于痕迹^[5]的特征识别方法。然而加工特征的多样性以及彼此间的相互交叉,会导致加工特征的破坏与以致分离,不利于制定完备的特征识别规则,因而识别交叉特征一直是加工特征识别领域公认的难题^[1-2,6]。此外,伴随特征的增多,整个三维模型的复杂度提高,在规则匹配阶段耗时严重^[7]。

利用大量标注样本以端到端的方式学习加工特征的存在模式,基于深度学习的特征识别方式避免了繁琐的规则制定过程并表现出更好的识别效果。该类方法将识别过程分解为特征类型的分类以及特征定位两个任务,根据两个任务的执行顺序可细分为两阶段方法与一阶段方法^[8]。两阶段方法采用单一加工特征数据集训练特征分类模型,随后基于预定义的规则将多特征模型分解为多个子模型,最后逐一识别。FeatureNet^[9]是第1个基于深度学习的方法,它将3D模型体素化并使用3D卷积网络来分类单特征模型,然后采用分水岭分割算法对三维模型进行分解。Ning等^[10]使用了类似的基于体素的方法,利用三维模型的凹边对多特征模型进行分解。Wang等^[11]基于B-rep提取面属性与面邻接图,采用图神经网络进行单个特征识别,并基于特征底面的深度信息设计交叉特征分解方法。Yao等^[12]采用点云表示具有单个特征的三维模型,并利用PointNet++^[13]实现单特征分类,然后利用点云法向量和凹边设计交叉特征分解策略。然而,两阶段方法依赖几何的连续性或者拓扑的邻接性设计启发规则实现特征的定位,仍难处理特征交叉导致的特征分离问题。

一阶段方法将特征定位亦纳入网络的学习过程中,Shi等^[8,14]提出了基于多视图方法的一阶段特征识别方式,同步实现特征分类以及特征包围盒预测任务,在多特征模型加工特征识别上表现出比FeatureNet更好的识别准确率。然而,基于包围盒的特征定位方式仅能实现特征的粗略定位,不能准确识别某加工特征具体包含的面,不便于后续的工艺规划与制造过程。Zhang等^[7]提出基于实例分割的加工特征识别方法—ASIN (associatively segmenting and identifying network)。ASIN包含语义分割、实例分组、底面识别3个分支,依次实现面的语义类型预测,以是否共加工特征为准则的面-面邻接矩阵预测以及加工特征底面预测。然而,ASIN采用点云数据表示三维模型,忽略了模型的拓扑关系,识别效果有待提高。当前,基于模型边界表示的特征识别方

法如CADNet^[15]、Hierarchical CADNet^[16]以及UV-Net^[17],使用图表达三维模型,以面邻接图作为基本的图结构,图的节点用于表示三维模型的面,图的边对应三维模型的边,并赋予相应的面、边属性。这种表达方式储存了模型的拓扑与几何信息,相较于点云、多视图、体素等欠缺拓扑信息的表示方法展示了更高的特征分类精度。然而,该类方法仅实现面级语义分类问题,不能进行特征的定位。

为提高特征交叉下加工特征识别能力,本文受益于基于边界表示的特征识别方法在特征分类任务上表现,提出加工特征实例分割方法Brep3pNet。该方法基于面点云数据与面邻接图等构成三维模型图数据表示,采用图神经网络以及基于度量学习思想的概率位置编码方法进行加工特征实例分割,实现特征的分类与定位。此外,深度学习依赖数据集。尽管当前存在如ABC、MCB等真实CAD零件数据集,但有别于零件分类任务,加工特征以三维模型拓扑面为基本单元,粒度细致,因此标注成本高。基于已有文献,当前用于加工特征识别领域的多特征数据集多为由立方体经过布尔操作而得,形状相对单一。为此,本文合成6 001个带实例标注的回转类零件,用于丰富加工特征领域数据的多样性并验证所提方法的有效性。

1 加工特征实例分割方法

1.1 方法概述

Brep3pNet的命名揭示了所提出方法的特征(B-rep, point cloud, probabilistic positional embedding),总体框架见图1。对于输入的STEP格式的三维模型,首先提取与每个面相关联的点云数据、面几何中心数据,模型的面邻接图(face adjacency graph, FAG)以及每个边的曲线类型和凹凸性信息,构成模型的图数据表示;然后将图数据输入到融合点云学习网络和图神经网络的编码器Brep1pNet中学习几何与拓扑信息,得到三维模型每个面的高维嵌入向量;之后将此嵌入向量分配给语义分割分支和实例分组分支,分别用于确定三维模型每个面的加工特征语义类别,以及每个特征实例中包含的面。为评估实例分组分支生成的实例质量,设计ScoreNet分支预测生成实例与真实实例的重叠度,形成实例质量得分。根据实例质量得分设计了贪心算法对得分较高且重叠度大的实例预测进行非极大抑制(non-maximal suppression, NMS)以得到精简的加工特征实例结果。最后,根据实例中各面的语义概率确定特征类型,实现加工特征的实例分割。

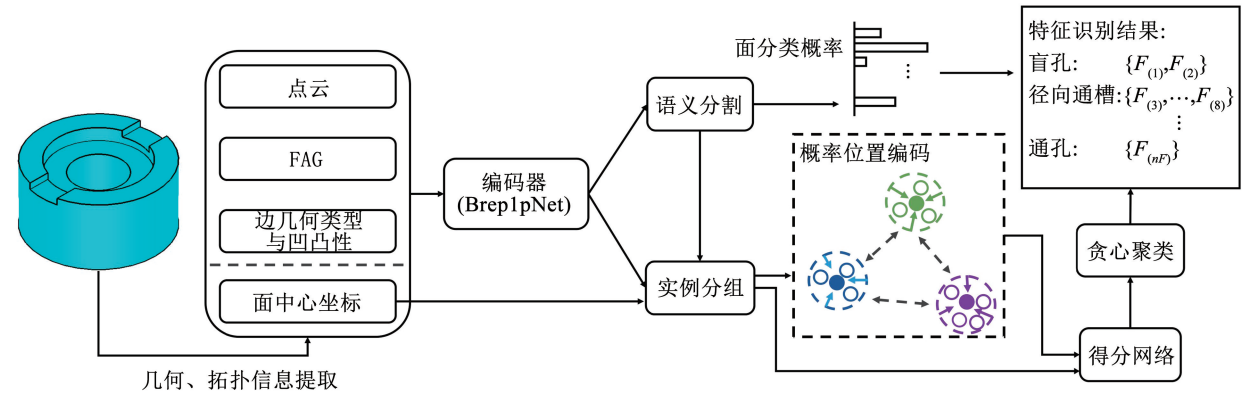


图 1 Brep3pNet 加工特征识别框架

Fig. 1 Framework of the proposed Brep3pNet for machining feature recognition

1.2 三维模型的图数据表示

边界表示是现代 CAD 软件主流的三维实体表示方法,它包含了重建此三维模型所需的所有几何和拓扑信息,涉及顶点、边、面、环、实体等多拓扑层次及其关联关系。由于边界表示的复杂性和非结构化,很难直接作为神经网络的输入^[8]。

鉴于面和边是加工特征识别任务的核心和关键

辅助信息,本文构建了如图 2 所示的以面邻接图为基础图结构的三维模型图数据表示。面邻接图上的节点对应三维模型的面,以面上采样的点云数据作为其属性;面邻接图上的边对应三维模型的边,以边的几何类型以及凹凸性作为其属性。相关数据的提取流程如下。

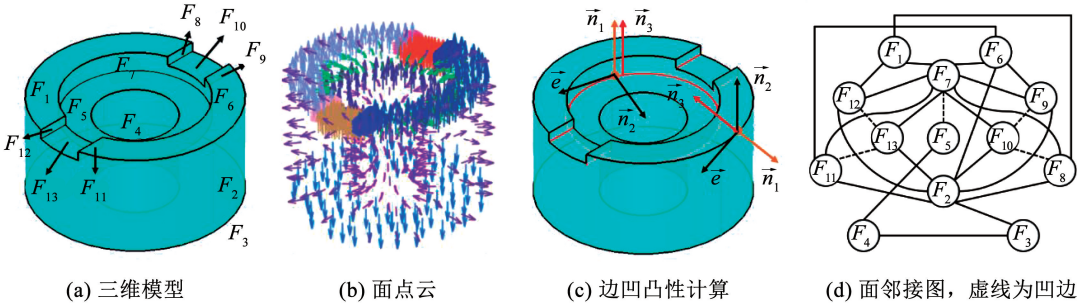


图 2 面点云、边凹凸性与面邻接图

Fig. 2 Face point cloud, edge concavity and FAG

1.2.1 点云数据提取

点云是三维实体常用的显式表达方式之一,蕴含丰富的几何信息。同时,点云作为计算机视觉领域的热门研究对象之一,积累了很多有益的研究成果,因此本文使用三维模型各面的点云数据作为三维模型几何信息表示的载体。三维曲面由 u 和 v 两个参数控制,这两个参数构成了曲面的 UV 空间。然而,在 UV 空间内采样点云存在如下两个问题:一是尽管基于 B-rep 可以方便得到曲面二维 UV 空间的矩形边界,但它覆盖了额外的区域,需要附加判断来确定采样点是否在表面上;二是由于非线性变换,UV 空间内采样的均匀性不能代表三维空间下的均匀性。考虑采样点的有效性与均匀性,本文采取“网格化+下采样”的策略实现三维模型各面的点云采样,前者保证误差允许范围内采样的点在面

上,后者进一步引入采样点间的距离约束,保证点云可均匀覆盖模型的面。在实现上,本文采用 pythonocc^[18]集成的三角剖分算法产生网格数据,并采用 Open3d^[19]对三角形网格执行泊松盘下采样,最终为每个面采样 $nP = 128$ 个点作为其几何表示,采样的面点云数据见图 2(b)。

1.2.2 面属性邻接图与边属性

面邻接图是基于图的特征识别方法的重要数据组成,它以三维模型的边为媒介表示三维模型面间的邻接关系,蕴含三维模型的拓扑信息。通过在 B-rep 上执行拓扑遍历,可以获得每个面所包含的边以及每个边关联的面,因此可以方便地构造 FAG。

虽然加工特征识别的单位为面,但边对识别起重要作用。本文利用边的两个属性:边的曲线类型

以及边的凹凸性。曲线类型包含直线、圆弧等,为几何信息。边的凹凸性则反映共有此边的两个面之间的拓扑连接方式。另外,凹边通常伴随着加工特征的存在,因此边的凹凸性是基于图的方法和一些基于两阶段学习的方法的关键属性。边的几何类型包含在 B-rep 之中,可直接获取,而凹凸性则是由与之关联两面所张成的二面角判断,若此二面角大于 180° 则为凸边,小于 180° 则为凹边。具体可采用如下等价的步骤计算:

- 1) 选择边上一点,并获取该点在两个面上的单位法向量 $\vec{n}_1$ 与 $\vec{n}_2$,方向指向三维模型外部。
- 2) 选定两面中的任一为参考,根据其法向量(如 $\vec{n}_1$)与右手定则,确定边的切向量 $\vec{e}$ 。
- 3) 计算 $\vec{n}_3 = \vec{e} \times \vec{n}_2$,以及 $\cos = \vec{n}_3 \cdot \vec{n}_1$ 。
- 4) 若 $\cos > 0$ 则判定此边为凹边,否则视为凸边。

图2中直观地总结了上述步骤。图2(c)为上述步骤的计算过程,其中边的切向量 $\vec{e}$ 由 $\vec{n}_1$ 确定,图2(d)为图2(a)所示三维模型的 FAG,虚线边表示凹边。此外,还可观察到因水平径向通槽的存在,模型的 FAG 从仅有两个阶梯孔时的6顶点环形图

变得更加复杂。且因与孔相交,该槽在空间上被分成两部分并在拓扑上发生分离,而确定此槽所包含的6个面正是本文所要解决的特征定位问题。

1.3 三维模型编码器

为学习三维模型中几何与拓扑信息,本文提出 Brep1pNet 充当三维模型面级的编码器,其主要包含面间共享的点云学习主干网络、多层含残差连接的图神经网络以及融合三维模型全局信息的聚合层模块,具体的网络结构见图3。给定含 nF 个面的三维模型及其图数据表示, Brep1pNet 通过点云学习网络将各面点云编码为 128 维向量,随后通过聚合层对各面的编码向量进行平均池化,获得模型全局几何表示,将其与各面向量拼接,获得更新的图节点向量矩阵 X_p ; 每条边的曲线类型和凹凸性则分别随机初始化为 4 维的可学习的向量,拼接成边属性 $E = \{e_{i,j}\} \subset \mathbb{R}^8$ 。之后采用多层含残差连接的图神经网络进行几何与拓扑信息的融合,实现面编码的更新,经过类似的全局池化与聚合操作得到最终三维模型面级编码 X_e ,即图的节点向量矩阵。

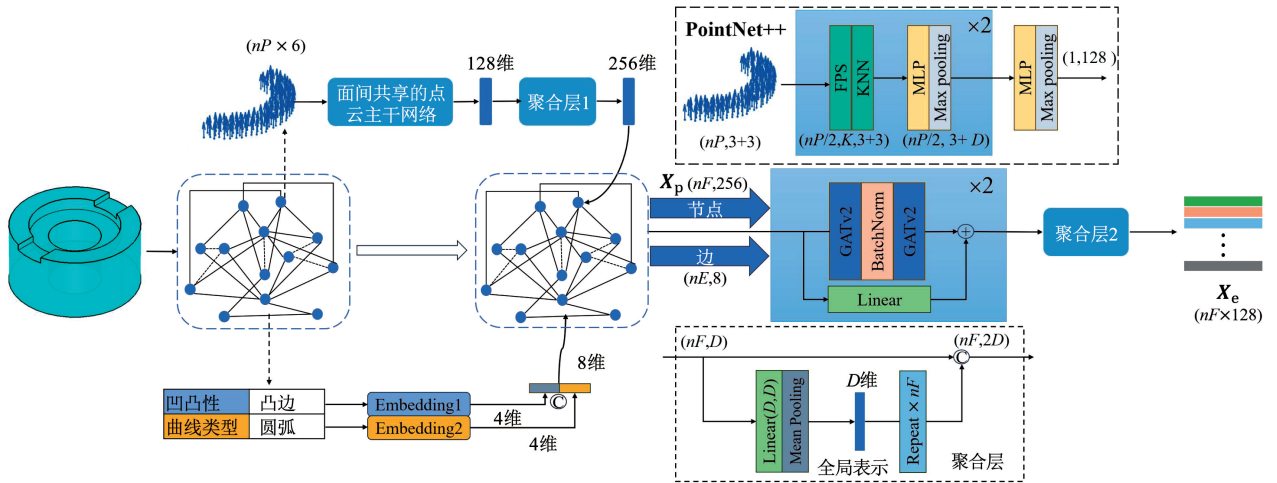


图3 编码器 Brep1pNet 的网络结构

Fig. 3 Architecture of Brep1pNet

本文采用点云学习中广泛使用的 PointNet++ 作为默认的主干网络。PointNet++ 是考虑点云局部信息的层次化模型,采用最远点下采样(farthest point sampling, FPS)选择点云的局部中心,并通过围绕中心的近邻查找构造局部区域,随后采用多层感知机(multilayer perceptron, MLP)和最大池化学习点云置换不变的特征表示。考虑到三维模型各面面积存在较大差异,为获取点的近邻点集,本文部署 K-近邻算法(K-nearest neighbor, KNN)代替官方固定半径的球查询方法。由于相邻面可分属相同或不

同的加工特征实例,本文采用基于注意力机制的 GATv2^[20]作为上述图神经网络。GATv2 采用式(1)~(3)进行消息传递,不同于另一种广泛使用的 GAT^[21](不同之处为式(4)),GATv2 实现了动态注意力,其所得注意力得分的相对大小与查询向量无关,因而有更好的特征提取能力。

$$x_i = \sum_{j \in N(i) \cup \{i\}} \beta_{i,j} \cdot Wx_j \quad (1)$$

$$\beta_{i,j} = \frac{\exp[\alpha(x_i, x_j)]}{\sum_{k \in N(i) \cup \{i\}} \exp[\alpha(x_i, x_k)]} \quad (2)$$

$$\alpha(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{a}^T \cdot \text{LeakyReLU}(\mathbf{W} \cdot [\mathbf{x}_i \| \mathbf{e}_{i,j} \| \mathbf{x}_j]) \tag{3}$$

$$\alpha(\mathbf{x}_i, \mathbf{x}_j) = \text{LeakyReLU}(\mathbf{a}^T \cdot [\mathbf{W}\mathbf{x}_i \| \mathbf{W}_e\mathbf{e}_{i,j} \| \mathbf{W}\mathbf{x}_j]) \tag{4}$$

式中： $\mathbf{x}_i$ 为图 3 中面邻接图第 i 个节点的节点向量，以第 1 层 GATv2 为例， $\mathbf{x}_i$ 即为 $\mathbf{X}_p$ 的第 i 行； $\mathbf{e}_{i,j}$ 为上述 8 维边属性， $\mathbf{W}$ 、 $\mathbf{a}$ 分别为待学习的权重矩阵与注意力向量。

表 1 列举了图 3 中包含的 4 层 GATv2 以及所用 PointNet++ 的 3 个 MLP 输入输出维度。其中，

表 1 编码器 Brep1pNet 的网络参数

Tab. 1 Parameters of Brep1pNet

子模块	网络层	输入特征尺寸	输出特征尺寸
PointNet++	MLP(3+3,32,32,64)	$(nF, 3+3, nP/2, K)$	$(nF, 64, nP/2, K)$
	MLP(3+64,64,64,128)	$(nF, 3+64, nP/4, K)$	$(nF, 128, nP/4, K)$
	MLP(3+128,256,128,128)	$(nF, 3+128, 1, nP/4)$	$(nF, 128, 1, nP/4)$
第 1 层 GNN 块	GATv2(256,64,4)	$(nF, 256)$	$(nF, 64)$
	GATv2(64,64,4)	$(nF, 64)$	$(nF, 64)$
第 2 层 GNN 块	GATv2(64,64,4)	$(nF, 64)$	$(nF, 64)$
	GATv2(64,64,4)	$(nF, 64)$	$(nF, 64)$

1.4 语义分割分支

语义分割分支为加工特征分类器,用于预测每个面所属的加工特征类型或概率,本文采用多层感知机对其进行建模,具体网络结构见图 4。

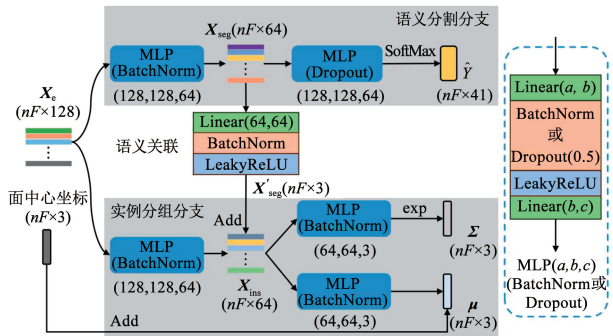


图 4 语义分割与实例分组分支结构

Fig. 4 Architecture of semantic segmentation and instance grouping branches

考虑到所提出的回转类零件加工特征数据集存在语义类别的不平衡问题,本文采用 Focal Loss^[22]作为训练阶段损失函数,见式(5)。相比于式(6)的交叉熵损失函数,Focal Loss 将模型输出的分类概率加权到损失函数中,相对地提高难分类样本的损失权重,适合处理样本不平衡问题。

$$L_{\text{sem}} = - \sum_{i=1}^{nC} y_i (1 - \hat{y}_i)^\gamma \ln \hat{y}_i \tag{5}$$

MLP 由卷积核大小为 1×1 的二维卷积(Conv2d)、批量归一化层、ReLU 层组成。以 MLP(3+64,64,64,128) 为例说明,该 MLP 有 3 个 Conv2d 层,各 Conv2d 的输入通道为 67、64、64,输出通道为 64、64、128,输入通道的前 3 维表示点的三维(x, y, z)坐标,层与层间含批量归一化与 ReLU 激活函数;GATv2(256,64,4)则表示其输入通道为 256,输出通道为 64,注意力头数为 4;本文每个面的 $nP = 128$,KNN 近邻数 $K = 24$ 。

$$L_{\text{sem}} = - \sum_{i=1}^{nC} y_i \ln \hat{y}_i \tag{6}$$

式中： nC 为加工特征的类别总数， y_i 为属于第 i 类的真实概率,值为 0 或 1； $\hat{y}_i$ 为属于第 i 类的概率预测值， γ 为折扣因子,用于调节各样本的重要性权重,在本文中设置 $\gamma = 2$ 。

1.5 基于概率位置编码的实例分组

加工特征由三维模型的面构成。对于一个含 nF 个面的三维模型,令 $\mathbf{F} = \{F_1, F_2, \dots, F_{nF}\}$ 表示三维模型面集合, $\mathbf{G} = \{G_1, G_2, \dots, G_m\}$ 表示模型所含加工特征的集合。对于每个加工特征 $G_i \in \mathbf{G}$, G_i 为面集 $\mathbf{F}$ 的子集,且 $\mathbf{G}$ 构成 $\mathbf{F}$ 的一个划分,满足如下两个性质:

- 1) $G_i \cap G_j = \emptyset, \forall i \neq j$ 。
- 2) $\cup_i G_i = \mathbf{F}$ 。

实例分组对应加工特征的定位任务,目标是获得三维模型中各加工特征对应的面集,即获得面集 $\mathbf{F}$ 的划分 $\hat{\mathbf{G}} = \{\hat{G}_1, \hat{G}_2, \dots, \hat{G}_{m'}\}$,以实现 $\mathbf{G}$ 的估计。为方便对应,本文将面 F_i 所在的加工特征记为 $G_{(i)}$ 。

1.5.1 概率位置嵌入及相似度度量

深度度量学习是常用且高效的实例分组方法,该方法基于同一实例下表示的相似性学习一个映射

函数(在此为神经网络),将组成实例的各基本单元(如图像实例分割中的像素,以及加工特征识别任务中的面)映射到可测成对相似度或距离的嵌入空间中。不同于 ASIN 与 AAG^[23] 中采用的高维向量,本文受益于文献[24],采用概率位置嵌入策略,将三维模型各面编码为考虑空间位置的三元高斯分布 $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, 其概率密度如下:

$$p(x) = \frac{1}{\sqrt{(2\pi)^n \det(\boldsymbol{\Sigma})}} \exp\left\{-\frac{1}{2}(x - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(x - \boldsymbol{\mu})\right\} \quad (7)$$

本文取 $n=3$, $\boldsymbol{\mu} \in \mathbf{R}^3$ 。为便于计算,取 $\boldsymbol{\Sigma}$ 为正定对角矩阵,由 3 个正分量决定。

巴氏距离 (Bhattacharyya distance)^[25] 常用于度量两概率分布 $p(x)$ 、 $q(x)$ 间距离(式(8)),其计算依赖于式(9)定义的巴氏核,相比于另一常用的概率间距离度量方法——KL 散度 (Kullback-Leibler divergence) 而言,具有对称性。

$$D_B(p, q) = -\ln K_B(p, q) \quad (8)$$

$$K_B(p, q) = \int_{\Omega} \sqrt{p(x)} \sqrt{q(x)} dx \quad (9)$$

式中: D_B 为巴氏距离, K_B 为巴氏核。

考虑到巴氏核取值为 $[0, 1]$, 本文采用原始巴氏距离度量同一实例下的概率嵌入距离,并定义下式度量不同实例下概率嵌入距离为

$$D_B^-(p, q) = -\ln(1 - K_B(p, q)) \quad (10)$$

给定面 F_i, F_j 的概率位置嵌入为 $\mathbf{f}_i \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ 与 $\mathbf{f}_j \sim N(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$, 其巴氏核的解析表达如下:

$$K_B(\mathbf{f}_i, \mathbf{f}_j) = \beta_{i,j} \times \exp\left\{-\frac{1}{8} \boldsymbol{\Delta}_{i,j}^T \boldsymbol{\Sigma}_{i,j}^{-1} \boldsymbol{\Delta}_{i,j}\right\} \quad (11)$$

其中:

$$\beta_{i,j} = \left(\frac{\det(\boldsymbol{\Sigma}_{i,j})}{\sqrt{\det(\boldsymbol{\Sigma}_i \boldsymbol{\Sigma}_j)}}\right)^{-\frac{1}{2}}$$

$$\boldsymbol{\Delta}_{i,j} = \boldsymbol{\mu}_i - \boldsymbol{\mu}_j$$

$$\boldsymbol{\Sigma}_{i,j} = \frac{\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j}{2}$$

对于一个加工特征实例 G_i , 令 g_i 表示此加工实例的对应的概率编码,可将其视为实例 G_i 内各面概率位置嵌入的平均,即 $g_i \sim N(\bar{\boldsymbol{\mu}}_i, \bar{\boldsymbol{\Sigma}}_i)$, 基于独立性假设, $\bar{\boldsymbol{\mu}}_i, \bar{\boldsymbol{\Sigma}}_i$ 的计算如下:

$$\bar{\boldsymbol{\mu}}_i = \frac{1}{|G_i|} \sum_{j \in G_i} \boldsymbol{\mu}_j \quad (12)$$

$$\bar{\boldsymbol{\Sigma}}_i = \frac{1}{|G_i|^2} \sum_{j \in G_i} \boldsymbol{\Sigma}_j \quad (13)$$

运算符 $|\cdot|$ 表示集合中元素个数。然而式(13)

不能很好表征实例中心的含义。假设一个加工特征实例的每个面都被编码为标准高斯分布 $N(\mathbf{0}, \mathbf{I})$, 显然更希望整个实例的编码仍为标准高斯分布, 因此本文实际采用下式计算 $\bar{\boldsymbol{\Sigma}}_i$:

$$\bar{\boldsymbol{\Sigma}}_i = \frac{1}{|G_i|} \sum_{j \in G_i} \boldsymbol{\Sigma}_j \quad (14)$$

1.5.2 网络结构与损失

实例分组分支的网络结构见图 4 所示。考虑到相同实例有相同语义, 本文对实例分组任务进行语义关联, 将语义特征 $\mathbf{X}_{\text{seg}}$ 投影到实例空间获得 $\mathbf{X}'_{\text{seg}}$, 求和计算实例级的特征向量 $\mathbf{X}_{\text{ins}}$, 随后通过两个独立的 MLP (BatchNorm) 层得到方差以及融合坐标信息的均值。

为获得高质量的嵌入表示, 提高加工特征的定位能力, 结合上述巴氏距离, 本文基于度量学习思想的 Double-Hinge Loss^[26] 设计实例损失。

$$L_{\text{ins}} = L_{\text{var}} + L_{\text{pull}} + L_{\text{push}} + \alpha L_{\text{reg}} \quad (15)$$

其中:

$$L_{\text{var}} = \frac{1}{nF} \sum_{i=1}^{nF} D_B(\mathbf{f}_i, \mathbf{g}_{(i)})$$

$$L_{\text{pull}} = \frac{1}{nF} \sum_{i=1}^{nF} \frac{1}{|G_{(i)}|} \sum_{j \in G_{(i)}} [D_B(\mathbf{f}_i, \mathbf{f}_j) - \delta_1]_+$$

$$L_{\text{push}} = \frac{1}{nF} \sum_{i=1}^{nF} \frac{1}{nF - |G_{(i)}|} \sum_{j \in F \setminus G_{(i)}} [D_B^-(\mathbf{f}_i, \mathbf{f}_j) - \delta_2]_+$$

$$L_{\text{reg}} = \frac{1}{nF} \sum_{i=1}^{nF} \ln \det(\boldsymbol{\Sigma}_i)$$

运算符 $[x]_+ = \max(x, 0)$ 。 $L_{\text{var}}, L_{\text{pull}}$ 分别用于拉近相同实例下各面在嵌入空间下的距离, 而 L_{push} 负责将不同实例下各面的概率嵌入推远, L_{reg} 则是为了防止方差过大产生无意义的高斯分布而增加的正则项^[24]。在本文, 设置 $\delta_1 = \delta_2 = 0.3, \alpha = 0.001$ 。

1.6 得分网络

通过本文提出的概率位置编码方法, 可以获得一个表征面实例嵌入距离的矩阵 $\mathbf{K}_B = \{K_{B,(i,j)}\} \in \mathbf{R}^{nF \times nF}$, 其中 $K_{B,(i,j)} = K_B(\mathbf{f}_i, \mathbf{f}_j)$ 。 $K_{B,(i,j)}$ 接近于 1, 表示面 F_i, F_j 属于同一个实例; 若其接近于 0, 则面 F_i, F_j 属于不同实例, 因此可通过设置阈值 K_τ 的方式获得实例 0-1 掩码矩阵 $\mathbf{M}^{\text{pre}}$ 。在模型训练阶段本文设置 K_τ 为 0.3, 与 δ_1 一致, 在验证与测试阶段将其放松中至 0.35。

$$\mathbf{M}^{\text{pre}} = (\mathbf{K}_B > K_\tau) \quad (16)$$

式中 $\mathbf{M}^{\text{pre}}$ 的每行代表以面 F_i 为锚点预测的加工特征实例。由于加工特征不只包含一个面, 因而真实的加工特征实例会对应 $\mathbf{M}^{\text{pre}}$ 的某几行, 但这几行并

不一定相同,因此有必要评估上述冗余实例的预测质量,以指导最终加工特征实例的获得。

令 $\tilde{G} = \{\tilde{G}_1, \tilde{G}_2, \dots, \tilde{G}_{nF}\}$ 表示以三维模型各面为锚点的加工特征实例预测,其中 $\tilde{G}_i = \{k: M_{i,k}^{\text{pre}} = 1\} \subset F$ 。为评估实例质量,本文部署如图 5 所示的 ScoreNet 分支为 $\tilde{G}$ 预测分数 $\hat{S} = \{\hat{s}_1, \hat{s}_2, \dots, \hat{s}_{nF}\}$,并以预测实例 $\tilde{G}_i$ 同其对应的真实实例 $G_{(i)}$ 间的集合相似度作为真实分数进行监督学习。图 5 中的 GNN 为单层的 GATv2,对应的图邻接矩阵为 M^{pre} ,以图 4 中实例成组分支的中间变量 X_{ins} 作为图的节点向量矩阵。

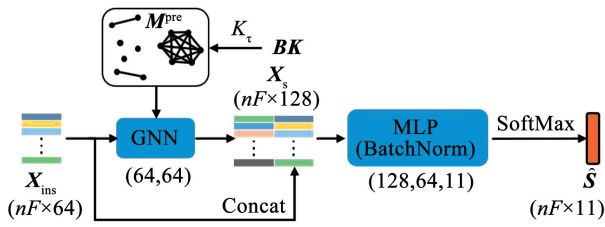


图 5 得分网络结构

Fig. 5 Architecture of ScoreNet

集合相似度采用集合间 IoU (intersection over union) 值(亦称 Jaccard 相似系数),对于集合 A 与 B 其 IoU 采用式 (17) 计算。考虑到每个加工特征的面数有限,如式 (18) 本文以 0.1 为间隔离散真实得分,将得分预测任务建模为类别数为 11 的多分类问题,并采用式 (6) 所示的交叉熵损失作为该分支损失函数。

$$\text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (17)$$

$$s_i = \text{IoU}(\tilde{G}_i, \tilde{G}_{(i)}) / 0.1 \quad (18)$$

1.7 推理过程

经过 Brep3pNet 的前向过程,可获得每个面的加工特征语义类型预测 $\hat{Y}$,关联到每个面的特征实例 $\tilde{G}$ 及其得分 $\hat{S}$ 。本文部署如下的贪心算法步骤去除 $\tilde{G}$ 中冗余实例,获取为最终加工特征预测 $\hat{G}$ 及其特征类型。

Step1 初始化剩余面集 $F_{\text{rest}} \leftarrow \{1, 2, \dots, nF\}$ 与最终加工特征实例 $\hat{G} = \emptyset$ 。

Step2 根据得分网络预测结果对各面进行降序排列,获得排列后得分及面索引: $S', I \leftarrow \text{Sort}(\hat{S})$ 。

Step3 选择剩余得分最大的面以及关联的特征面集 $G: i \leftarrow I[1]; G \leftarrow \tilde{G}_{(i)} \cap F_{\text{rest}}$ 。

Step4 将特征面集 G 添加到加工特征实例 $\hat{G}$ 中,并对 G 中各面语义概率求和,选择求和后概率最大的特征语义作为此加工特征的类型。

Step5 更新剩余面集及当前得分: $F_{\text{rest}}, I \leftarrow F_{\text{rest}} \setminus G, I \setminus G, S' \leftarrow \{\hat{s}_k, \text{for } k \in I\}$ 。

Step6 遍历剩余所有面 $j \in I$, 对于得分 $\hat{s}_j > \tau_1$ 的面获取其特征实例 $G' \leftarrow \tilde{G}_{(j)} \cap F_{\text{rest}}$, 如果 $\text{IoU}(G, G') > \tau_2$, 则对面的得分进行减半抑制: $\hat{s}_j \leftarrow \hat{s}_j / 2$ 。

Step7 更新得分并进行降序排序: $S' \leftarrow \{\hat{s}_k, \text{for } k \in I\}, S', I' \leftarrow \text{Sort}(S'), I \leftarrow \{I[k], \text{for } k \in I'\}$ 。

Step8 重复 Step3 ~ 7 至 $F_{\text{rest}} = \emptyset$, 输出最终特征实例 $\hat{G}$ 。

该方法依照实例得分大小决定最终加工特征实例,对于分数较高且重叠度大的一对实例,本文进行非极大抑制处理,即对其中分数较小者的得分进行减半。在算法中,通过阈值 τ_1, τ_2 衡量分数和重叠度的大小以决定是否进行抑制,本文设置 $\tau_1 = \tau_2 = 0.6$ 。

2 结果与讨论

本文使用 pytorch 1.10.2 以及 pytorch geometric 2.0.4 搭建 Brep3pNet 模型,使用搭载 AMD EPYC 7543 32-Core CPU 及 NVIDIA RTX A40 GPU 的云服务器进行模型训练,并在搭载 Intel(R) i7-8700K CPU 的本地计算机上进行识别测试。

2.1 数据集与评价指标

2.1.1 方形零件数据集

加工特征识别领域的多特征数据集主要包含 MFCAD、MFCAD++ 以及 MFInstSeg,此三者均基于方形毛坯通过去除部分材料合成。MFCAD 由 Cao 等^[15]提出,覆盖 16 种平面类加工特征,共含 15 488 个 CAD 样本,其中每个样本包含 1 ~ 6 个加工特征。由于 MFCAD 不包含非平面加工特征如常用的孔,Colligan 等^[16]拓展了 MFCAD 的生成方式并提出了 MFCAD++ 数据集,此数据集覆盖由 FeatureNet 所定义的 24 类加工特征,共有 59 655 个 CAD 样本,每个 CAD 样本包含 3 ~ 10 个加工特征。不过, MFCAD 与 MFCAD++ 并不包含加工特征实例标注,仅有 CAD 模型各面的加工特征类型标注,因此 Wu 等^[23]基于 MFCAD++ 数据集创建工具,提出含实例标注的 MFInstSeg 数据集,其三维模型类似 MFCAD++,覆盖 24 类加工特征,共约 60 000 个样本。MFCAD 与 MFCAD++ 部分样本见图 6。

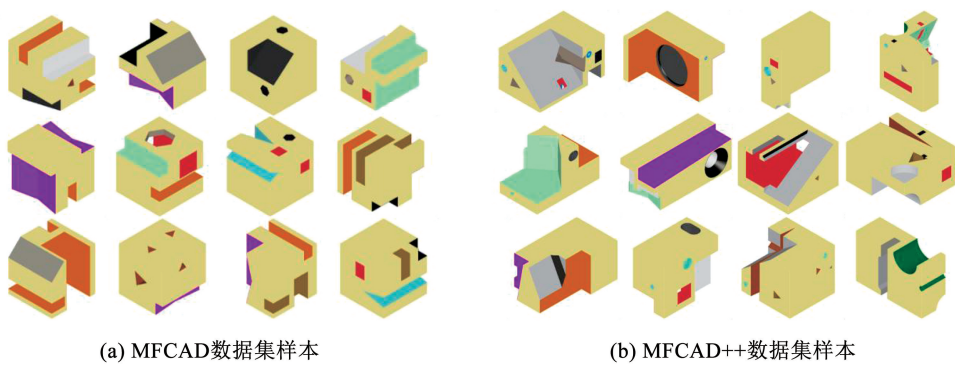


图 6 MFCAD 与 MFCAD++ 中部分三维模型

Fig. 6 Partial 3D models from MFCAD and MFCAD++ datasets

2.1.2 回转类零件数据集创建

上述加工特征识别的数据集,形状相对单一,制约数据驱动类特征识别方法的应用。本文基于参数化建模思想,以 pythonocc 作为建模工具构建带加工特征实例标注的回转类零件数据集。基于文献[9]

所定义的 24 类常见加工特征并结合特征在回转类零件的轴向与径向两类位置,本文定义了如图 7 所示的 37 类加工特征,涉及 4 类回转型基体,包含圆柱基体、圆锥基体和球面基体以及由前三者基于布尔加操作合成的多层基体。

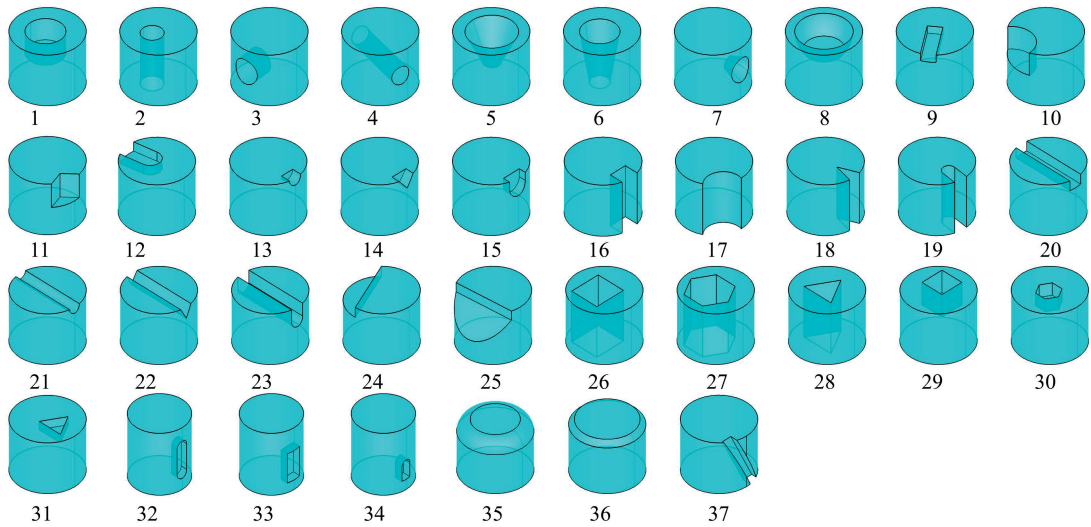


图 7 所提数据集包含的加工特征类型

Fig. 7 Machining features involved in the proposed dataset

以单基体零件为例,说明 CAD 样本的生成与标注过程。本文固定基体形状的最大半径为 100,并基于此约束随机获得基体与加工特征的几何参数。在确定零件基体后,依次添加 1~6 个加工特征以形成最终三维模型样本。每添加一个加工特征对应一次布尔减操作,即从当前模型去除相应的形状以形成此特征。pythonocc 具有追踪模型构建历史的特性,在每次布尔操作后,可以获得新增的面集,由此可形成当前加工特征的实例标签,即加工特征实例所含面的集合。相应的,可以获得面的删除与变动情况以更新历史加工特征实例。另外,本文采用文献[14]所述的多体检测,以防止 3D 模型失效。最

终,本文生成 6 001 个三维模型样本,其部分三维模型样本见图 8。



图 8 所提数据集的部分 3D 模型

Fig. 8 Partial 3D models from the proposed dataset

2.1.3 评价指标

为验证 Brep3pNet 特征识别效果,本文采用文献[16]定义的面语义分类准确率(记为 A_{sem})评估语义分割分支面语义分类结果;定义 G_{IoU} 如式(19),表示所有加工特征实例中被正确定位的实例占比,用于表征特征定位准确率。上述两个指标是对 Brep3pNet 中各分支的独立评估,可细致展示网络各模块的效果。考虑到加工特征识别是分类与定位的统一,使用文献[23]中的方法,以特征定位准确且类别正确的样本作为阳性样本,计算特征识别的准确率(记为 A_{FR})以及 F1 score (记为 F_1),以此评估模型最终识别结果。本文保留验证集 A_{sem} 与 G_{IoU} 之和的最优模型用于测试集测试。

$$G_{IoU} = \frac{|\{i:\hat{G}_i \in G, i = 1, 2, \dots, |\hat{G}|\}|}{|\hat{G}|} \quad (19)$$

2.2 对比实验与分析

2.2.1 语义分割任务下实验结果与分析

本文部署 Brep3pNet 的编码器 Brep1pNet 及语义分割分支在加工特征语义分割数据集 MFCAD 以及 MFCAD++ 上开展识别实验并与其他先进方法比较。为对比公平,两数据集均采用官方的数据划分获得训练集、验证集和测试集。同文献[16],本文使用 Adams 优化器在 MFCAD 数据集上模型训练 350 轮,在 MFCAD++ 数据集上模型训练 100 轮,批量大小均为 128。其中,训练 1 轮表示训练集全部样本通过神经网络。本文对学习率进行阶梯式衰减,在训练过程的前 50 轮保持初始学习率为 0.001,随后在 MFCAD 数据集上以 50 轮为周期减半学习率,在 MFCAD++ 数据上以 10 轮为周期减半学习率。所提方法在两数据集上的识别结果见表 2、3。不同方法使用的三维模型的表示不同,对于 DGCNN^[27]、PointNet++ 以及 MeshCNN^[28] 而言,其识别的几何单元分别为点、点以及三角面片,对应的三维模型面级的特征分类结果由面所关联的基本几何单元投票确定。所列举的 PointNet++、Hierarchical CADNet 等位于表格第 1 栏的实验结果摘录于文献[16],表格第 2 栏是在上述超参数设置下的实验结果。

从表 2、3 可以看出,由于两数据集模型复杂度不同,各识别方法在 MFCAD++ 数据集上的面分类准确率均有下降。尽管如此,本文方法在 MFCAD 数据集和 MFCAD++ 数据集上分别以最少的模型参数量实现最高 99.99% 和 99.09% 的面分类准确率,表明所提方法具有轻量、高效的优势。

表 2 MFCAD 数据集下不同模型语义分类精度及参数量

Tab. 2 Semantic classification accuracy and parameter number of different models in the MFCAD dataset

模型	$A_{sem}/\%$		参数量/M
	面级	几何单元级	
DGCNN	90.99	71.00	0.53
PointNet++	91.35	94.95	1.42
CADNet	99.95		0.53
MeshCNN	99.89	98.52	2.29
UV-Net	99.95		1.23
Hierarchical CADNet	99.90		6.60
ASIN	98.94		4.36
Brep3pNet	99.99		0.33

表 3 MFCAD++ 数据集下不同模型语义分类精度及参数量

Tab. 3 Semantic classification accuracy and parameter number of different models in the MFCAD++ dataset

模型	$A_{sem}/\%$		参数量/M
	面级	几何单元级	
DGCNN	85.98	71.00	0.53
PointNet++	85.88	94.95	1.42
Hierarchical CADNet	97.73		9.76
ASIN	91.62		4.37
UV-Net	98.85		1.37
Brep3pNet	99.09		0.33

在用于比较的方法中,UV-Net 的表现最接近本文方法。UV-Net 同 Brep3pNet 均采样面的点集学习面几何信息,并使用 FAG 以及边属性进一步融合三维模型拓扑信息,因此识别结果相近。不过,两种方法在面、边属性的数据表示上存在差异。具体而言,Brep3pNet 的点集为无序的点云,利用 PointNet、PointNet++ 等点云学习网络,而 UV-Net 在面的 UV 空间等距采点,形成有序的 UV 网格,利用 2D 卷积网络学习。本文使用边的几何类型和凹凸性表示边信息,而 UV-Net 对各边一维参数空间下等距采点,通过 1D 卷积学习边特征。相较而言,没有融合拓扑信息的 ASIN 在两个数据集上的识别结果均低于 Hierarchical CADNet、UV-Net 以及 Brep3pNet,特别是在样本更复杂的 MFCAD++ 数据集上仅实现 91.62% 的面分类准确率,相较于 Brep3pNet 有 7.47% 的下降。图 9 展示了 Brep3pNet 与 ASIN 训练过程前 50 轮验证集下面分类历史最优准确率曲线,可以明显看出融合拓扑信息的 Brep3pNet 相比于 ASIN 有更高效的学习能力,表明融合拓扑信息的图方法在加工特征识别上的优越性。

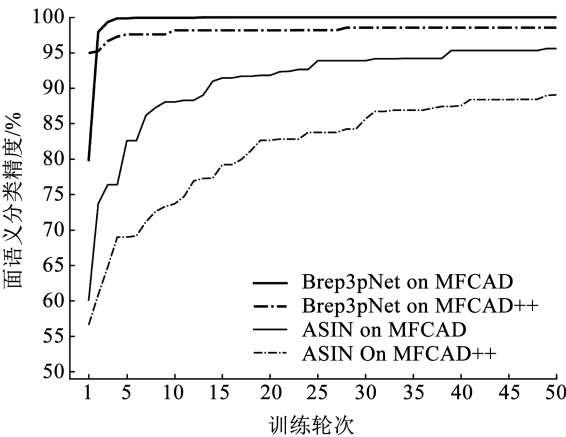


图 9 训练前 50 轮内验证集累计最优语义分类准确率曲线

Fig. 9 Cumulative optimal semantic classification accuracy curve on the validation set for the first 50 epochs of the training phase

2.2.2 MFInstSeg 实验结果与分析

本文在 MFInstSeg 数据集部署完整的 Brep3pNet 模型用于加工特征实例分割,采用官方默认的数据集划分方式获得训练集、验证集与测试集,损失函数见式(20),并使用 Adams 优化器进行模型训练。同文献[23],设置训练轮次为 100,批量大小为 256。在训练过程的前 50 轮保持初始学习率为 0.001,随后以 10 轮为周期减半学习率。表 4 展示了 Brep3pNet, ASIN 以及 AAGNet 在 MFInstSeg 数据集上的识别结

果,其中 ASIN 的训练配置同文献[23],AAGNet 使用官方公布的模型。

$$L_{\text{total}} = L_{\text{seg}} + L_{\text{ins}} + L_{\text{score}} \quad (20)$$

AAGNet 同 Brep3pNet 类似均为基于模型边界表示的加工特征识别方法,相比于 ASIN 均使用 FAG 融合三维模型拓扑信息,因此在模型轻量化与加工特征识别性能上面均优于 ASIN。由于 MFInstSeg 数据集同 MFCAD++ 数据集采用相似的生成方式获得,其中三维模型相近,因而 Brep3pNet 在 MFInstSeg 数据集上的语义分类结果与在 MFCAD++ 上的结果相近。AAGNet 在三维模型面与边数据表示方面继承自 UV-Net,使用二维 UV 网格与一维点序列表示面与边的几何信息。此外,AAGNet 增加面的几何类型(平面、圆柱、锥面等)、中心与面积等几何属性,因而在语义分割任务上实现更高的 99.52% 分类正确率。基于本文所提概率位置编码以及用于特征评估的 ScoreNet, Brep3pNet 在特征定位方面获得 97.73% 的 G_{IoU} ,高于 AAGNet。归功于特征定位上的优势,即使输入信息量少于 AAGNet, Brep3pNet 实现 96.95% 的特征识别正确率和 96.71% 特征识别 F_1 ,表现出更好的加工特征识别能力。此外,Brep3pNet 无明显增加推理时间,相对于部署 Mean shift 聚类的 ASIN 更高效。

表 4 MFInstSeg 数据集下不同模型特征识别结果对比

Tab. 4 Comparison of feature recognition results on MFInstSeg dataset

模型	$A_{\text{sem}}/\%$	$G_{\text{IoU}}/\%$	$A_{\text{FR}}/\%$	$F_1/\%$	平均推理时间 GPU/s	参数量/M
ASIN	91.55	63.60	61.07	60.78	0.220 3	6.14
AAGNet	99.52	96.68	96.26	96.49	0.017 1	0.44
Brep3pNet	99.05	97.73	96.95	96.71	0.017 2	0.39

2.2.3 回转类零件数据集模型对比结果

本文依据零件的面数与特征数对生成的回转类零件数据集进行划分,其中训练集、验证集、测试集的样本量分别为 5 121、400、480,对应的实例数与面数的统计结果见表 5。对于 Brep3pNet,本文设置初始学习率为 0.000 5,每隔 50 轮学习率减半,训练损失函数为式(20)。AAGNet 与 ASIN 则采用官方推荐的初始学习率及衰减模式,各模型的训练迭代轮次均为 280,批量大小为 16,实验结果见表 6。

表 5 面数及加工特征实例数统计

Tab. 5 Statistics of faces and machining feature instances

数据划分	平均实例数	平均面数	样本大小
训练集	7.76	16.14	512 1
验证集	7.81	16.23	400
测试集	7.81	16.28	480
总计	7.77	16.16	600 1

表 6 回转类零件数据集下不同模型特征识别结果

Tab. 6 Comparison of feature recognition results on rotary part dataset

模型	$A_{\text{sem}}/\%$	$G_{\text{IoU}}/\%$	$A_{\text{FR}}/\%$	$F_1/\%$	平均推理时间 GPU/s	参数量/M
ASIN	82.03	40.88	40.29	18.12	0.761 7	5.26
AAGNet	98.36	95.55	94.82	92.72	0.066 4	0.42
Brep3pNet	98.05	97.21	95.98	93.57	0.068 2	0.38

对比 MFInstSeg 数据集,本文数据集特征类型更多且样本量有所下降,因而识别难度更大,对应的识别表现均有下降。从表 6 的结果来看,Brep3pNet 实现 97.21% 的 G_{IoU} ,相较于 AAGNet 有更好的定位能力,因此最终在特征识别准确率和特征 F_1 两个特征级指标上表现更好,同时两模型识别效率接近,与 MFInstSeg 数据集的实验结论一致。

ASIN 的识别结果显得格外差。除缺少关键的拓扑信息外,一个重要原因是因三维模型面数不同产生的数据不平衡。ASIN 以面的点云数据表达三维模型,数据结构为 $[nF, nP, 6]$ 的 3 维张量,需采用补 0 的方式引入“缺失面”为所有三维模型统一面数以形成 $[三维模型数, nF, nP, 6]$ 的 4 维张量,实现小批量训练。这将引入额外的“缺失”特征类型,在三维模型面数差异较大的情况下会产生明显的 数据不平衡,进而导致模型训练不稳定最终识别结果差的问题。Brep3pNet 和 AAGNet 使用图结构表示三维模型,多个三维模型的 FAG 邻接矩阵将按照对角线排列以合成一张大图,以此实现批量训练,因而受面数影响更小。本文合成的数据集因为包含单基体模型和更复杂的多基体两种,数据集面数的均值约为 16 而最大面数为 66,三维模型的面数波动相对于单立方基体的 MFInstSeg 等数据集更大,因而 ASIN 的表现力衰减明显。考虑真实零件将更加复杂多样,模型的复杂度差异将更明显,因此基于图的三维模型表示方式可更好适应面级实例分割的零件加工特征识别任务。

2.3 消融实验

2.3.1 整体架构消融

为探究网络输入、网络结构对加工特征识别结果的影响,本文在所提数据集上进行消融实验。其中,所探究的网络输入包含 FAG 与边属性,以及用

于概率位置编码的面中心坐标;网络结构包含点云主干网络以及是否进行语义联合。特别地,为进一步探究概率位置编码的有效性,本文使用与概率位置编码相同变量数的 6 维向量以及同 ASIN 采用的 64 维向量作为对比,分别记为 EC-6D 和 EC-64D,相应的使用余弦核替换巴氏核,具体公式如下:

$$C_k(f_i, f_j) = \frac{1 + \cos(f_i, f_j)}{2}$$

(21)

式中: f_i 为面 F_i 经实例分组合分支编码得到的 6 维或 64 维特征向量。

消融实验结果见表 7。表 7 中结果显示,在没有 FAG 输入情况下,即未融入拓扑信息及边属性信息,模型的分类精度下降达 13%,而不引入概率位置编码,在相同的变量维度下实例分组的表现力下降超过 50%,即使增加至 ASIN 与 AAGNet 所采用的 64 维,也仍有约 23% 的差距,因此可认为融入从边界表示中的拓扑信息以及部署的概率位置编码策略是 Brep3pNet 的核心。其次对结果有更重要影响的是边属性信息,亦是增加拓扑信息对改进识别结果的反应。无面位置信息的概率编码与无语义联合下的实例分组结果相近,在 G_{IoU} 指标上有约 0.8% 的下降,可以说面中心坐标与语义信息的融入对特征定位有促进作用。在面几何信息提取方面,本文使用 PointNet^[29] 与 PointMLP^[30] 代替 PointNet++ 进行实验对比。结果表明关注点云局部信息的层次化模型如 PointNet++ 与 PointMLP 相比于 PointNet 有更好的特征识别结果。进一步对比上述其他输入或网络结构下的识别结果,可以看出点云主干网络的选择对识别结果的影响更小,但由于 PointNet++ 相比于 PointMLP 计算效率更高,因此被选为推荐的点云学习主干网络。

表 7 输入及网络结构消融结果

Tab. 7 Input and architecture ablation results

模型结构		$A_{sem}/\%$	$G_{IoU}/\%$	指标变化量		相对重要性
				$A_{sem}/\%$	$G_{IoU}/\%$	
默认配置		98.05	97.21			
Brep	- wo FAG	84.99	93.99	- 13.06	- 3.22	高
	PointNet	97.25	96.86	- 0.80	- 0.35	低
	PointMLP	97.43	97.29	- 0.62	+ 0.08	
3P	EC-6D	97.36	43.64	- 0.69	- 53.57	高
	EC-64D	97.68	74.22	- 0.37	- 22.99	

表 7 (续)

模型结构		$A_{sem}/\%$	$G_{IoU}/\%$	指标变化量		相对重要性
				$A_{sem}/\%$	$G_{IoU}/\%$	
其他	Position - <i>wo</i> 面中心坐标	97.54	96.54	-0.51	-0.67	低
	- <i>wo</i> 边属性	96.88	95.63	-1.17	-1.58	中
	- <i>wo</i> 语义关联	97.55	96.59	-0.50	-0.62	低

2.3.2 点云分辨率

本文使用点云进行模型的几何表示,因此每个面所采样点云的大小是关联几何信息的重要超参数。由于 PointNet++ 为包含点云下采样且考虑局部信息的层次化模型,因此其层数、点云数 nP 、KNN 的近邻数 K 会有制约。本文设置 PointNet++ 的层数为 2,包含两次点云下采样,使用 (nP,K) 的组合为 $(32,8)$ 、 $(64,16)$ 及 $(128,24)$ 共 3 组实验以探究点云数对特征识别的影响,其实验结果见表 8。实验结果表明,随着点云数的增加,Brep3pNet 语义以及实例分割的结果均有所提高,与文献[6]的结论一致。在本文生成的数据集中,点云大小从 32 增加到 128,Brep3pNet 的语义分类准确率提高 0.41%,实例分组准确率提高 0.24%,对比整体架构消融中的边属性、面中心坐标、语义融合以及 PointNet 等输入或模型结构,点云分辨率的影响相对更小。在 MFInstSeg 数据集中,点云大小从 32 增加到 128,语义分类准确率提高 0.16%,实例分组准确率提高 0.32%,同样改善较小。

表 8 不同点云大小下特征识别结果

Tab.8 Feature recognition results at different point cloud sizes				
点云分辨率 (nP,K)	回转类零件数据集		MFInstSeg 数据集	
	A_{sem}	G_{IoU}	A_{sem}	G_{IoU}
(32,8)	97.64	96.97	98.89	97.41
(64,16)	97.88	97.15	99.00	97.53
(128,24)	98.05	97.21	99.05	97.73

2.3.3 图神经网络层数

GNN 层数对应 FAG 各节点可用于信息聚合的邻居阶数,是关联拓扑信息的重要超参数,因此本文进一步探究 GNN 层数对识别表现的影响。本文每 2 层 GNN 通过残差连接形成 GNN 块,因此层数成对递增。实验结果如图 10 所示,层数从 0~2,本质对应拓扑信息由无到有,识别效果特别是面语义分类精度明显提升,可以进一步说明面分类任务相对

于定位任务对拓扑信息更敏感;层数由 2 增大到 4 并不能明显增加识别结果,这可能与大多数加工特征内部邻接度数小的特点有关;而层数由 4 增大到 6 存在识别效果下降的趋势,这或许与三维模型本身,也就是一个面的 6 阶邻居近乎包含整个模型,以及多层 GNN 存在的过平滑现象相关联。

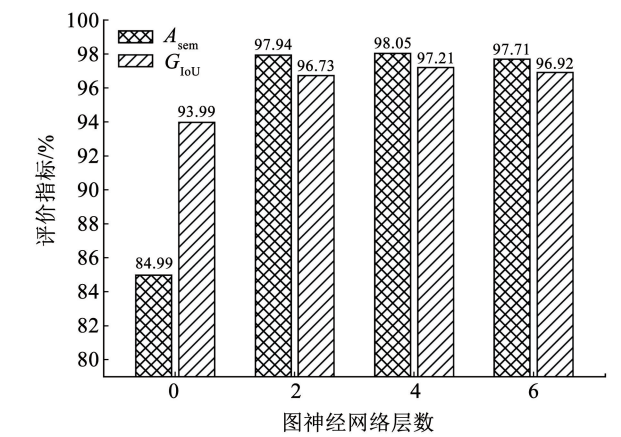


图 10 图神经网络层数对识别结果的影响

Fig. 10 Impact of GNN depth on feature recognition results

2.4 案例可视化

为进一步验证概率位置编码方法在特征定位任务上的有效性,本文对实例分组分支的输出进行可视化展示。将概率位置编码得到的三元高斯分布表示为 3 维椭球,椭球中心为高斯分布的均值,椭球的 3 个轴表示方差。相比于通用的高维向量编码方式而言,无需使用 T-SNE 或 KPCA(ASIN 采用的)等降维算法,即可直观反映为何 Brep3pNet 将某些面视为同一个加工特征。图 11 为所生成的回转类零件测试集中 3 个被正确识别(特征类型以及特征定位均正确)的模型可视化案例。这 3 个案例包含不同形式的特征交叉,存在着同一加工特征在几何空间和拓扑的分离现象如实例 Ins. b2、Ins. b5 等,但 Brep3pNet 仍可以将这些分离的面聚成一组。最直观的体现为同一实例内表示面高斯分布编码的椭球彼此重叠,不同实例间椭球分离。加工特征实例 Ins. a3 包含的两个面的椭球表示虽没有重叠,反映出两个高斯分布的均值存在较大差异,但由于方差

接近, Brep3pNet 依然可将其正确定位。图 12 为方形零件的特征识别可视化案例, 3 个零件来自文献[23], 共涉及 161 个面、20 个加工特征实例, 均被正确识别。就特征定位而言, 同一特征内各面的编

码得到的高斯分布高度重合, 与其他面明显区分, 且存在明显的相对位置关系, 可以说明本文所采用的概率位置编码方式可以有效利用加工特征空间位置的先验信息, 实现良好的特征定位效果。

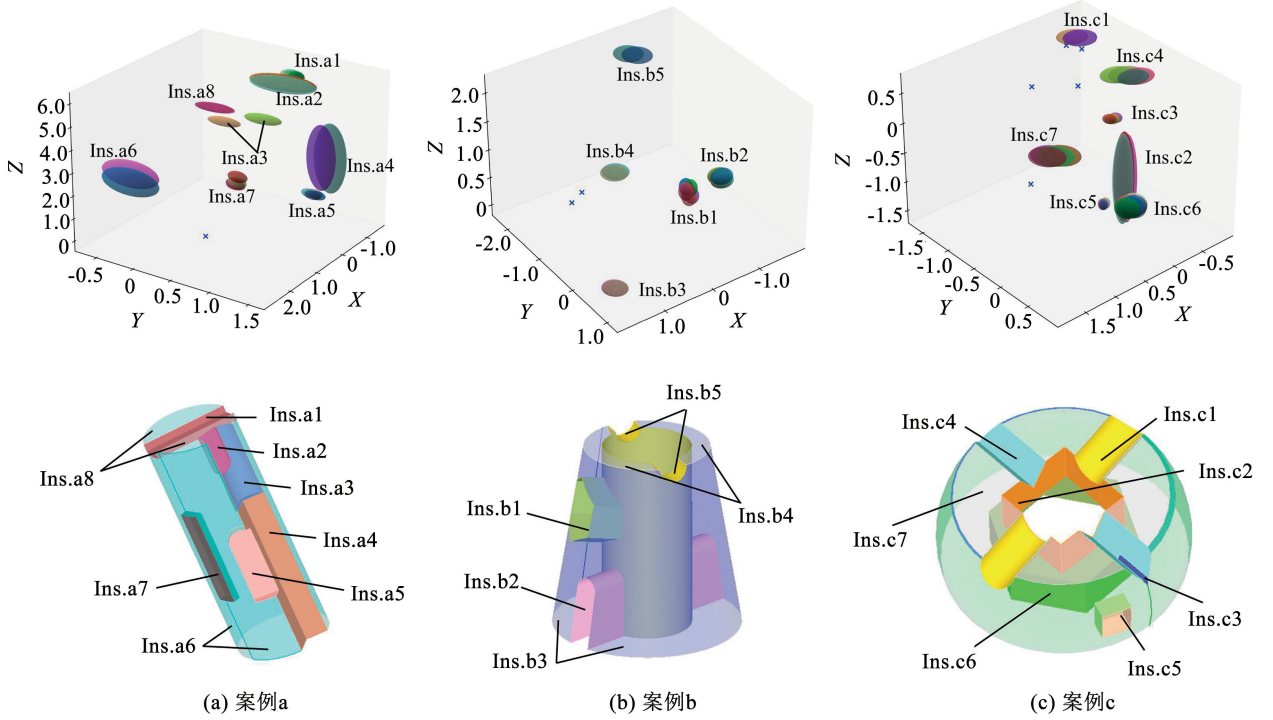


图 11 回转类零件特征识别与定位结果

Fig. 11 Feature recognition and localization results of rotary parts

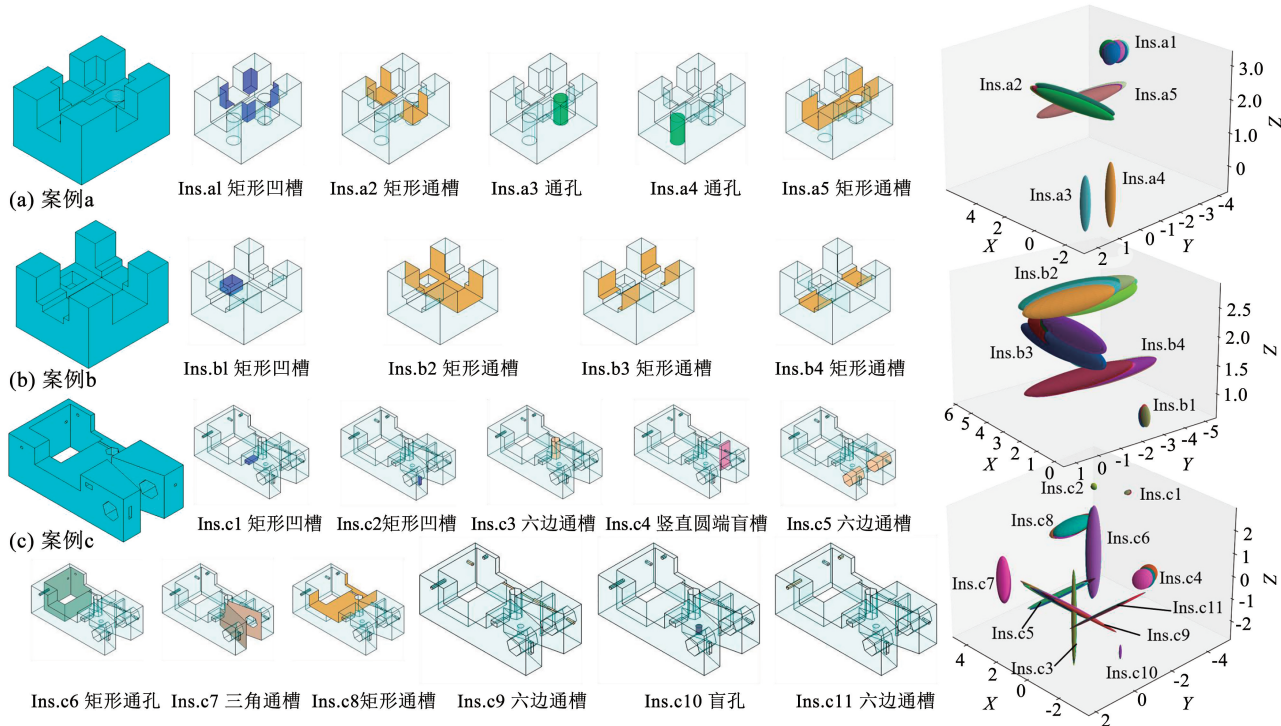


图 12 方形零件特征识别与定位结果

Fig. 12 Feature recognition and localization results of cuboid parts

最后,本文将所提方法应用于真实喷嘴组件的副喷口、主喷口以及旋流器,其特征识别结果见图 13。图 13 中 3 个模型共包含 69 个面(不考虑副喷口圆柱基底上的 2 个凹陷)以及 54 个实例,仅有 4 个面的语义分类错误以及 1 个实例的定位错误。对于错误预测的实例 Ins. a13, 其面的高斯分布间的均值差异大但方差差异小,虽类似实例 Ins. a3,但由于两个高斯分布间的巴氏核大于本文所定阈值 $K_r = 0.35$,因而识别错误。换言之,适当提高 K_r ,可以避免此次错误。此外,可发现 4 个语义分类错误均与

倒角或圆角相关:圆锥面 F. a2 与 F. b1 被错误预测为倒角;小圆柱面 F. a1 被错误预测成圆角;而邻接倒角的面 F. a3 更为特别,被错误归类为锥盲孔,这或许也导致实例 Ins. a13 的定位错误。锥孔、外锥面同倒角相比,几何形状上相似均包含圆锥面但几何尺寸不同,可进一步融合几何尺寸信息提高 Brep3pNet 特征识别能力。面 F. a1 表示的外圆面特征过小,不符合用于训练 Brep3pNet 的回转类数据集下的外圆面存在模式,泛化能力有待提高。

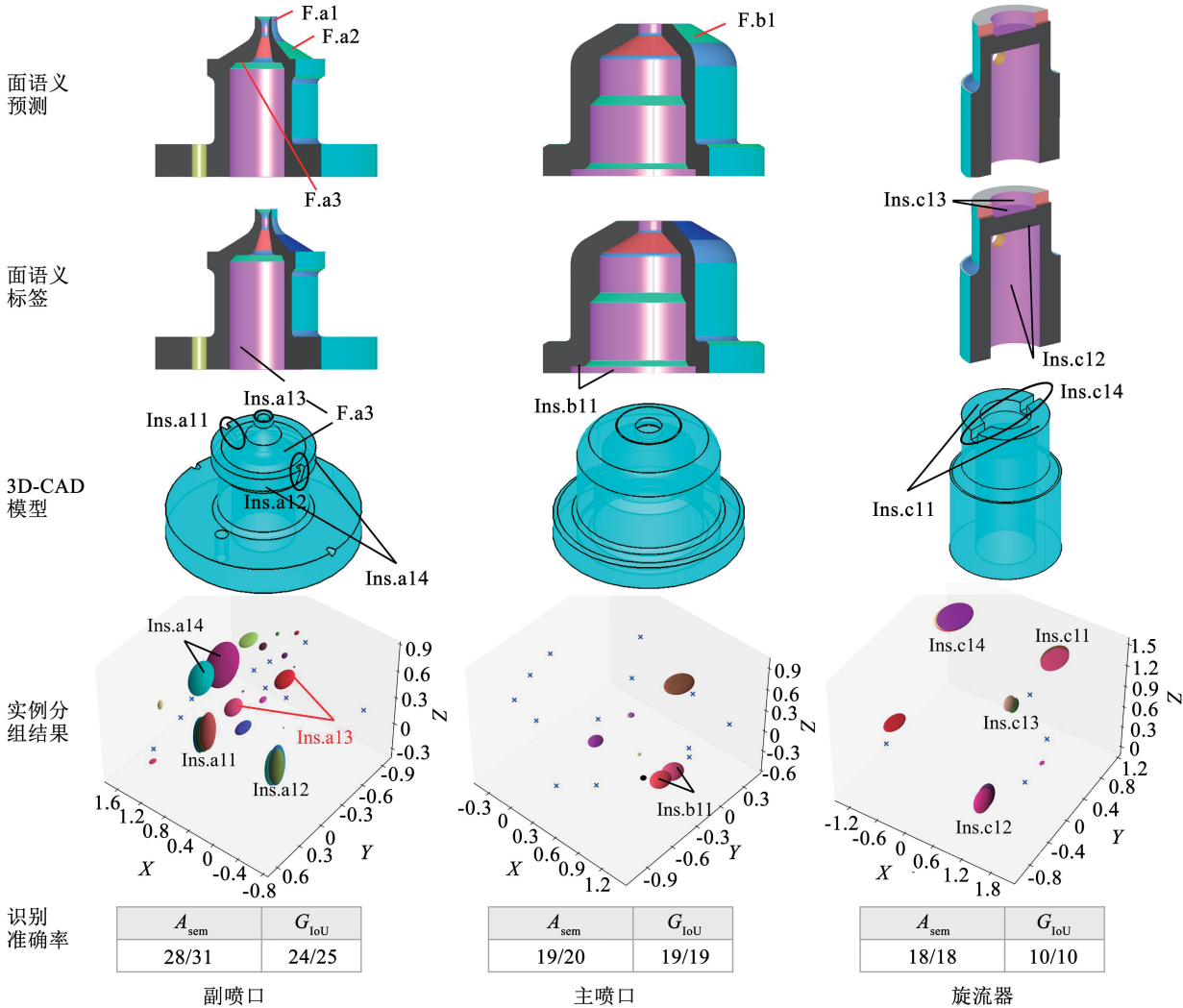


图 13 真实喷嘴组件的特征识别与定位结果

Fig. 13 Feature recognition and localization results of real nozzle acomponents

3 结论

1) 针对特征交叉下的加工特征识别问题,本文提出一种加工特征实例分割方法— Brep3pNet。通过提取面点云、边凹凸性与曲线类型以及面邻接图构建三维模型的结构化图表示,联合点云学习网络与图神经网络抽取模型几何与拓扑信息,同步完成特征分类与特征定位任务。此外,合成了含特征实

例级标注回转类零件数据集,缓解了当前加工特征识别领域数据集多由方形零件组成造成的样本模式单一、应用受限问题。

2) 利用面邻接图构建三维模型的图表示可有效融合模型的拓扑信息,在 MFCAD 和 MFCAD++数据集上以最少的模型参数量实现 99.99% 和 99.09% 的最优面语义分类精度,相对于同样采用面点云作为几何信息的 ASIN 方法, Brep3pNet 收敛更

快、识别准确率更高。消融实验可进一步表明,此拓扑信息的重要性高于点云分辨率、点云学习网络的选择。

3) 提出概率位置编码的加工特征定位方法在 MFInstSeg 与所生成的回转类数据集上实现 97.73% 与 97.21% 的特征定位准确率,优于 ASIN 方法与输入信息量更多的 AAGNet。在同样网络结构下,对比 6 维与 64 维常用的向量式编码方法,可将特征定位准确率提高 53.57% 与 22.99%。概率编码的可视化显示各加工特征实例会保持相对的空间位置关系,说明所提概率位置编码方法可有效利用模型各面的空间位置信息,提高特征定位能力。

4) 尽管 Brep3pNet 在模型轻量化、识别效率以及效果方面展现了最优或接近最优的结果,但仍有局限。后续将进一步融合几何尺寸等信息提高三维模型表示能力,并将研究重心集中于提高模型识别更复杂零件加工特征的泛化能力。

参考文献

- [1] SHI Yang, ZHANG Yicha, XIA Kaishu, et al. A critical review of feature recognition techniques [J]. *Computer-Aided Design and Applications*, 2020, 17(5): 861. DOI: 10.14733/cadaps.2020.861-899
- [2] VERMA A K, RAJOTIA S. A review of machining feature recognition methodologies [J]. *International Journal of Computer Integrated Manufacturing*, 2010, 23(4): 353. DOI: 10.1080/09511921003642121
- [3] 刘雪梅, 贾勇琪, 陈祖瑞, 等. 缸体类零件加工特征识别方法 [J]. *计算机集成制造系统*, 2016, 22(5): 1197
- [4] LIU Xuemei, JIA Yongqi, CHEN Zurui, et al. Recognition of machining feature for engine cylinder blocks [J]. *Computer Integrated Manufacturing Systems*, 2016, 22(5): 1197. DOI: 10.13196/j.cims.2016.05.004
- [5] WOO Y, SAKURAI H. Recognition of maximal features by volume decomposition [J]. *Computer-Aided Design*, 2002, 34(3): 195. DOI: 10.1016/S0010-4485(01)00080-X
- [6] LI Haiyan, HUANG Yunbao, SUN Yuhang, et al. Hint-based generic shape feature recognition from three-dimensional B-rep models [J]. *Advances in Mechanical Engineering*, 2015, 7(4): 782. DOI: 10.1177/1687814015582082
- [7] POBOZNIAK J. Interacting manufacturing features in CAPP systems [M]//HAMROL A, CISZAK O, LEGUTKO S, et al, eds. *Lecture Notes in Mechanical Engineering*. Cham: Springer International Publishing, 2017: 249. DOI: 10.1007/978-3-319-68619-6_24
- [8] ZHANG Hang, ZHANG Shusheng, ZHANG Yajun, et al. Machining feature recognition based on a novel multi-task deep learning network [J]. *Robotics and Computer-Integrated Manufacturing*, 2022, 77: 102369. DOI: 10.1016/j.rcim.2022.102369
- [9] SHI Peizhi, QI Qunfen, QIN Yuchu, et al. Intersecting machining feature localization and recognition via single shot multibox detector [J]. *IEEE Transactions on Industrial Informatics*, 2021, 17(5): 3292. DOI: 10.1109/TII.2020.3030620
- [10] ZHANG Zhibo, JAISWAL P, RAI R. FeatureNet: Machining feature recognition based on 3D Convolution Neural Network [J]. *Computer-Aided Design*, 2018, 101: 12. DOI: 10.1016/j.cad.2018.03.006
- [11] NING Fangwei, SHI Yan, CAI Maolin, et al. Part machining feature recognition based on a deep learning method [J]. *Journal of Intelligent Manufacturing*, 2023, 34(2): 809. DOI: 10.1007/s10845-021-01827-7
- [12] YAO Xinhua, WANG Di, YU Tao, et al. A machining feature recognition approach based on hierarchical neural network for multi-feature point cloud models [J]. *Journal of Intelligent Manufacturing*, 2023, 34(6): 2599. DOI: 10.1007/s10845-022-01939-8
- [13] QI C R, YI Li, SU Hao, et al. PointNet++: deep hierarchical feature learning on point sets in a metric space [EB/OL]. 2017: 1706.02413. <https://arxiv.org/abs/1706.02413v1>
- [14] SHI Peizhi, QI Qunfen, QIN Yuchu, et al. Highly interacting machining feature recognition via small sample learning [J]. *Robotics and Computer-Integrated Manufacturing*, 2022, 73: 102260. DOI: 10.1016/j.rcim.2021.102260
- [15] CAO Weijuan, ROBINSON T, HUA Yang, et al. Graph representation of 3D CAD models for machining feature recognition with deep learning [C]//ASME 2020 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Virtual: [s. n.], 2020. DOI: 10.1115/DETC2020-22355
- [16] 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: 103226. DOI: 10.1016/j.cad.2022.103226
- [17] JAYARAMAN P K, SANGHI A, LAMBOURNE J G, et al. UV-Net: learning from boundary representations [C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville: IEEE, 2021: 11698. DOI: 10.1109/CVPR46437.2021.01153
- [18] PAVIOT T. Pythonocc-core: python package for 3D CAD [R]. BIM/PLM/CAM. Available online: <https://github.com/tpaviot/pythonocc-core>
- [19] ZHOU Qianyi, PARK J, KOLTUN V. Open3D: a modern library for 3D data processing [EB/OL]. 2018: 1801.09847. <https://arxiv.org/abs/1801.09847v1>
- [20] BRODY S, ALON U, YAHAV E. How attentive are graph attention networks? [EB/OL]. 2021: 2105.14491. <https://arxiv.org/abs/2105.14491v3>
- [21] VELIČKOVIĆ P, CUCURULL G, CASANOVA A, et al. Graph attention networks [EB/OL]. 2017: 1710.10903. <https://arxiv.org/abs/1710.10903v3>
- [22] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020, 42(2): 318. DOI: 10.1109/TPAMI.2018.2858826
- [23] WU Hongjin, LEI Ruoshan, PENG Yibing, et al. AAGNet: a graph neural network towards multi-task machining feature recognition [J]. *Robotics and Computer-Integrated Manufacturing*, 2024, 86: 102661. DOI: 10.1016/j.rcim.2023.102661
- [24] ZHANG Biao, WONKA P. Point cloud instance segmentation using probabilistic embeddings [C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville: IEEE, 2021: 8879. DOI: 10.1109/CVPR46437.2021.00877
- [25] JEBARA T, KONDOR R, HOWARD A. Probability product kernels [J]. *Journal of Machine Learning Research*, 2004, 5: 819
- [26] WANG Weiyue, YU R, HUANG Qiangui, et al. SGPN: similarity group proposal network for 3D point cloud instance segmentation [C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City: IEEE, 2018: 2569. DOI: 10.1109/CVPR.2018.00272
- [27] WANG Yue, SUN Yongbin, LIU Ziwei, et al. Dynamic graph CNN for learning on point clouds [J]. *ACM Transactions on Graphics (TOG)*, 2018, 38: 1. DOI: 10.1145/3326362
- [28] HANOCKA R, HERTZ A, FISH N, et al. MeshCNN [J]. *ACM Transactions on Graphics*, 2019, 38(4): 1. DOI: 10.1145/3306346.3322959
- [29] CHARLES R Q, HAO Su, MO Kaichun, et al. PointNet: deep learning on point sets for 3D classification and segmentation [C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE, 2017: 77. DOI: 10.1109/CVPR.2017.16
- [30] MA Xu, QIN Can, YOU Haoxuan, et al. Rethinking network design and local geometry in point cloud: a simple residual MLP framework [EB/OL]. 2022: 2202.07123. <https://arxiv.org/abs/2202.07123v2>