

题目

摘要

本文利用了 ComplexGen 框架，通过两个步骤解决了从点云输入到通过构建复形来重建并生成 CAD 模型的问题。第一个步骤是通过一个 CNN 网络的编码器处理输入点云，并将编码器的输出通过一个含有三通道的 Transformer 解码器来输出 CAD 模型的几何图元以及图元之间的拓扑关系。第二个步骤是通过一个全局优化方法以及几何提炼方法，从神经网络预测出的带概率的模型结构中恢复出一个 B-Rep 复形链，通过该复形链重建出 CAD 模型。最后，本文尝试了将 ComplexGen 框架中的全局优化方法迁移至与原文不同的数据集上进行操作。

关键词：复形链；全局优化；几何重建

1 引言

从一个无结构的点云中重建一个精确的有结构的 CAD 模型表示是一个十分基础，但又是一个有着广泛研究与商业应用的课题，像这样的工作一般也被称作逆向工程。以往的工作都尝试将点云先进行分割，然后将这些分割后的点云与不同几何类型或者参数的基础元素进行拟合，例如以网络学习为基础的将输入点云分割为不重合的区域并将这些分割后的点云与不同类型的基础面块相拟合 [6]，或是通过检测连接在线框上较为尖锐的点和边来对点云进行分割并拟合 [9]。纵然这些工作在各自的所重建的基础元素类型上都有着十分精确的结果，但还没有一项工作能够同时重建模型的面、边以及点，也就导致了重建工作的不完善。鉴于此种不完善的重建，建立一个有着一致结构和正确 CAD 模型拓扑的的重建工作十分重要。

ComplexGen 的框架全面重建了完整的模型结构，也就是说它能够同时重建出模型的点、边、面，以及这些图元之间的拓扑结构，这样的重建方法不仅本身是一个十分有诱惑力的重建目标，其本身也能增强 CAD 模型重建本身的可靠性。ComplexGen 的框架分成了两个部分：第一部分包含了从点云中检测图元以及它们之间的拓扑关系，其中图元的几何嵌入是一个回归任务而拓扑关系则是一个分类任务；第二部分，则是在拥有几何图元及其拓扑关系这样一个能够很好的约束 CAD 模型的互补信息的基础上，通过引入约束项来重建完整的 CAD 模型。

为了对这样一个全面的结构进行建模，ComplexGen 采用了边界表示方法 (B-Rep) 作为其基础结构。B-Rep 结构由不同图元 (点、边、面) 以及图元之间的拓扑结构组成。B-Rep 同时也定义了一个链复形，而链复形能够很好的表示出结构的约束，例如位于面的边界上的边一定形成一个闭合的环，这些约束必须满足才能进行结构有效的 CAD 模型重建。最后本文

还对 ComplexGen 中的全局优化尝试进行迁移，应用在大型 CAD 数据集 ABC 上 [5]，以期待在真实的 CAD 数据集上其能够生成正确的模型拓扑结构。

2 相关工作

2.1 CAD 重建

CAD 重建基于图元的 CAD 重建在工业应用上十分重要的应用，但是这个问题天生就会伴随着图元组合搜索和数据拟合的连续优化问题，因此现存的方法大致分为传统的专注在数值优化的方法，以及近期这些神经网络学习为驱动专注于克服数据庞大的复杂性的方法。

为了将图元面片与输入的离散点云或者多面体网格相拟合，早期有着像 RANSAC [12] 一样应用概率搜索或者解各种各样优化问题的方法 [2]。当然也有通过在图形结构的约束上应用全局优化来增强局部图元的拟合 [7, 10]。同时也有一些更先进的旨在通过深度学习的方法来解决以数据为驱动的 CAD 模型重建问题 [6, 13]。

2.2 基于 Transformer 的物体检测

物体检测与分割的流水线在近年有着巨大的变动——从以往的基于 CNN 的对输入图片中的锚点做出物体提议的方法 [4] 到现在通过涌现的方法基于输入的条件直接输出物体的绝对位置 [1]。上述所提到的基于机器学习的 CAD 模型重建方法都是传统的基于 CNN 先将输入点云中的点作为潜在的锚点进行分类，再将这些点合并后进行查找的检测方法。而 PolyGen [11] 则是使用了以 Transformer 为骨架的自回归方法来生成顶点的绝对位置以及他们对于面的归属。ComplexGen 则是在 DETR 的框架 [1] 的基础上构建直接检测图元类别的网络，当然在图元类别之外，ComplexGen 通过额外生成图元拓扑的方法增强了 CAD 模型生成的准确性与一致性。

此部分对课题内容相关的工作进行简要的分类概括与描述，二级标题中的内容为示意，可按照行文内容进行增删与更改，若二级标题无法对描述内容进行概括，可自行增加三级标题，后面内容同样如此，引文的 bib 文件统一粘贴到 **refs.bib** 中并采用如下引用方式。

3 本文方法

3.1 本文方法概述

如图1所示，ComplexGen 的框架是从无结构的点云输入中同时重建出 CAD 模型的顶点、边、面，以及这些图元之间的拓扑关系。

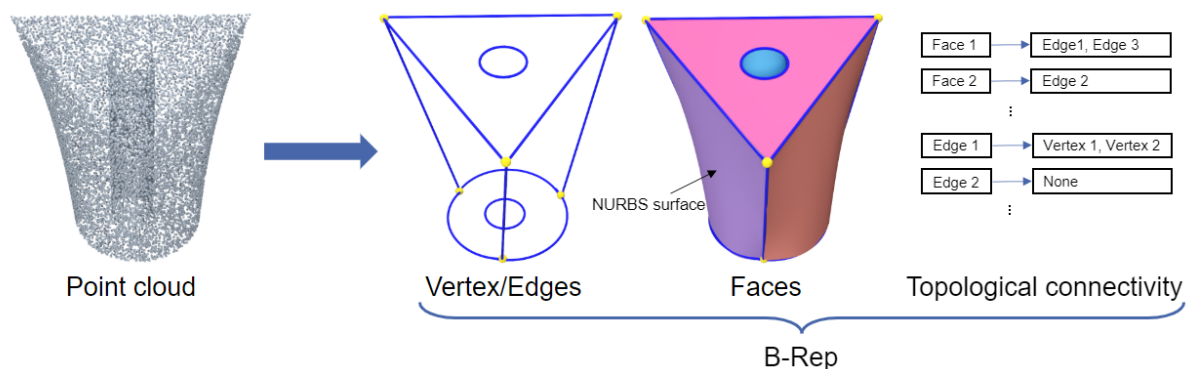


图 1. ComplexGen 输入与输出示意图

3.2 ComplexGen 管线

如图2所示，ComplexGen 管线大致分为两个子任务：对点云中模型的构成元素以及其拓扑的检测任务，以及几何细节的回归任务。其中第一个任务，即从点云中获得并检测出模型的图元与图元间的拓扑关系是通过一个 ComplexNet 来实现的，即管线的第 1 与第 2 阶段，而第二个子任务：几何细节的回归，也即复形提取与几何提炼则构成管线的第 3 与第 4 阶段。

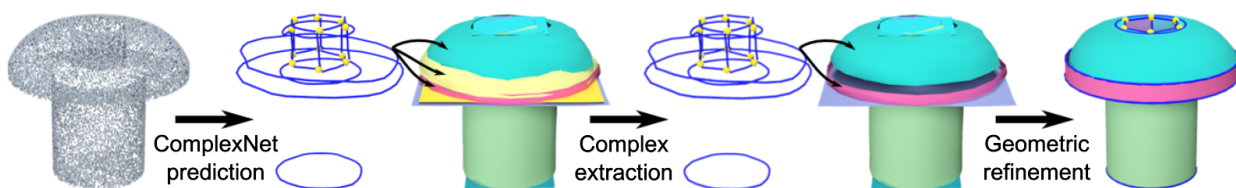


图 2. ComplexGen 生成有结构的 CAD 模型一共有 4 个阶段：1. 点云输入。2. 通过 ComplexNet 预测得到组成模型的基本元素（点、边、面）以及这些基本元素之间的拓扑关系。3. 通过全局优化技术解得这些基础元素能够使全局最优的存在性与连接性。4. 通过将第 3 步中得到的模型结构与初始的输入点云、相互约束关系以及预测得到的基本元素类型相拟合，以此来达到几何改善的目的，改善后的基本元素共同构成了一个合法的 B-Rep 模型

3.3 ComplexNet

如图3所示，ComplexNet 包含两个部分：一个稀疏的 CNN 编码器，该编码器的作用是从一个已经被离散化为一个稀疏的体素的点云中提取特征向量；一个拥有三个通道的以 Transformer 为骨架的解码器，该解码器旨在同生成模型的顶点、边、面以及这三种图元相互之间的拓扑关系。

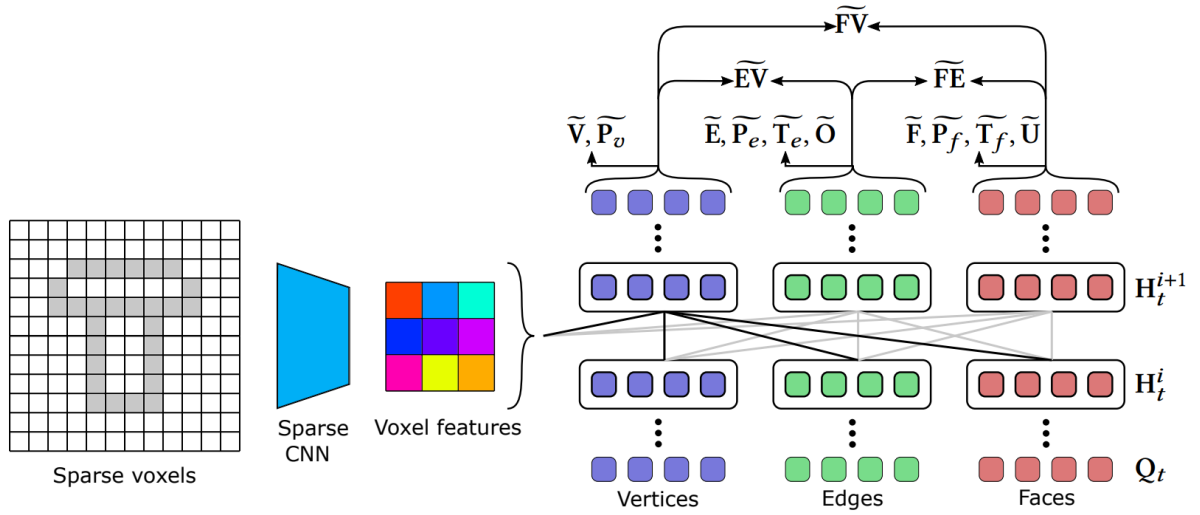


图 3. ComplexNet 结构。ComplexNet 训练的输出项有图元的合法性 $\tilde{V}, \tilde{E}, \tilde{F}$ ，此外边和面的闭合也需要进行相应的预测以满足后续优化的约束项需要，即 \tilde{O}, \tilde{U} ，二元拓扑项 $\tilde{EV}, \tilde{FV}, \tilde{FE}$ 也会同时输出

3.4 链复形提取

链复形提取的目标是恢复出一个在结构上是合法的流行的 B-Rep 模型，并且这个模型是尽可能的遵循上一阶段网络对于模型的图元与拓扑预测输出的。链复形的提取分为两个部分：非最大抑制和组合优化。

3.4.1 非最大抑制

在进行组合优化之前，需要先对一些冗余的相似图元进行移除，对于冗余图元的定义如下：1) 图元预测为合法 ($validness \geq 0.5$) 2) 存在一个 q' 并且 q' 的合法性也大于 0.5，且与 q 的拓扑结构相同，预测的图元形状也相同。那么对于冗余的图元，则直接将其合法性置为 0。ComplexNet 的预测结果为固定 100 个顶点，150 条边以及 100 个面，通过这种方法可是过滤掉一部分 ComplexNet 网络预测结果中大量冗余重复的图元，大大减少接下来组合优化的计算量，而对于需要迁移到的 ABC 数据集 [5]，这样的操作反而显得多余，然而 ABC 数据集中也存在一些特殊情况，需要对数据进行进一步的处理，该特殊情况也会在接下来进行详细阐述。

3.4.2 组合优化

复形结构的提取可以视作一个二元优化问题，其优化目标如下：

$$\begin{aligned}
& \max \quad wF_{topo} + (1 - w)F_{geom} \\
& \text{s.t.} \quad \begin{cases} \sum_i \mathbf{FE}[i, j] = 2\mathbf{E}[j], & j \in [N_e] \\ \sum_j \mathbf{EV}[i, j] = 2\mathbf{E}[i]\mathbf{O}[i], & i \in [N_e] \\ \mathbf{FE} \times \mathbf{EV} = 2\mathbf{FV} \end{cases} \\
& \begin{cases} \mathbf{FE}[i, j] \leq \mathbf{F}[i](\leq \sum_j \mathbf{FE}[i, j]), & \forall i, j \\ \mathbf{EV}[i, j] \leq \mathbf{V}[i] \leq \sum_k \mathbf{EV}[k, j], & \forall i, j \end{cases}
\end{aligned} \tag{1}$$

其中 $w = 0.5$ 为拓扑优化目标的权重, $[N_e]$ 为边的数量。前三个约束是一个合法的流形所必须满足的约束, 可以解释为:

- 一条边必定连接两条边。
- 一条封闭/开放的边, 必定被 0/2 个顶点所约束。
- 一个面的边界必定是一个封闭的环。

而剩下的两条约束同样也是提取复形时所必须满足的。如果一个面不存在, 则拓扑上不应该存在任何与其相连的边, 而如果一个面存在 (非球面), 则在拓扑上必定存在边约束其边界; 对于顶点也同理, 如果一个顶点不存在, 则拓扑上不应该存在与其相连的边, 而如果一个顶点存在, 则一定存在边与其相连, 或者说, 该点一定是由某些边相交而产生的。

此外为了度量优化的目标值, 拓扑目标与几何目标值的计算分别如下:

$$F_{topo} = \sum_{\mathbf{X}} w_{\mathbf{X}} (2\tilde{X} - 1) \cdot \mathbf{X} \tag{2}$$

$$F_{geom} = \sum_{\mathbf{X}} w_{\mathbf{X}} (2S_{\mathbf{X}} - 1) \cdot \mathbf{X} \tag{3}$$

其中 \mathbf{X} 为 $\{\mathbf{V}, \mathbf{E}, \mathbf{F}, \mathbf{O}, \mathbf{EV}, \mathbf{FV}, \mathbf{FE}\}$, 即图元自身存在的合法性性与代表图形图元拓扑结构的合法性, 且对于 $\{\mathbf{V}, \mathbf{E}, \mathbf{F}, \mathbf{O}\}$ 权重 $w_{\mathbf{X}} = 10$, 而对于二元量 $\{\mathbf{EV}, \mathbf{FV}, \mathbf{FE}\}$ 权重 $w_{\mathbf{X}} = 1$, 通过这种权重分配方式可以平衡这两种不同变量之间巨大的数量差异。此外 $S_{\mathbf{X}}$ 是一个度量拓扑拟合程度的量, 其计算公式如下:

$$S(a, b) = \exp\left(-\frac{d_{a,b}^2}{\epsilon^2}\right) \in (0, 1] \tag{4}$$

$$d_{a,b} = \frac{1}{|P_a|} \sum_{p_a \in P_a} \min_{p_b \in P_b} \|p_a - p_b\| \tag{5}$$

其中 $\epsilon = 0.1$ 是用来控制 S 的变化速率, $d_{a,b}$ 为两个不同类型图元之间采样点的平均距离, 且 a 的级别比 b 低 (如 a 类型为顶点而 b 类型为边)。

通过观察上述约束条件不难发现, 如果需要将链复形提取操作迁移至 ABC 数据集 [5] 中, 则需要对数据集本身的数据做一些处理, ABC 数据集中的图形存在一种不符合合法流形定义的特殊情况, 即圆柱面, 圆柱面是由一个 NURBS 曲面自相交所构成的, 因此会存在一条母

线，该条母线含有两条方向不同，但是图元类型、合法性以及拓扑都几乎一致的边，考虑到链复形提取的目标需要构建出合法流形的目标，因此需要删除该构成该母线的两条边，即将图元合法性与相对应的拓扑都置为 0。

3.5 几何提炼

经由上一步链复形的提取后，则可以获得图形的拓扑与各个符合拓扑与链复形约束图元的类型（例如：圆柱面，球面），文本将通过把输入点云与对应图元类型的几何表示相拟合以获得符合拓扑结构的图形。

如算法1所示，几何提炼部分被分为了两个部分，第一阶段会使用样条面片拟合除了球面与平面的所有类型的面片。而第二阶段，本文将把每一个面片转化为其预测的类型并依照该类型的约束再次与输入点云进行拟合。

Algorithm 1 Procedure of geometric refinement

Input: complex with solved topology and predicted geometric primitives, input points, $K_1 = 3, K_2 = 5$

Output: geometric primitives conforming to prescribed topology and input points

// First stage: fitting with spline patches

for $i \in [K_1]$ **do**

 fit each patch as spline (except plane, sphere) to input points, curves, and corners;

 fit each curve to patches and corners;

 fit each corner to curves and patches;

end for

// Second stage: fitting with typed primitive patches

for $i \in [K_2]$ **do**

 fit each typed patch to input points, curves, and corners;

 fit each curve to patches and corners;

 fit each corner to curves and patches;

end for

4 复现细节

4.1 与已有开源代码对比

本文参考了 ComplexGen 复形提取部分的代码结构，并以该结构重新编写了运行代码，源代码仅支持 ComplexNet 输出的文件格式，现代码可以支持处理后的 ABC 数据集中的数据，即对 ABC 数据集中 STEP 格式的文件先进行顶点、边与面的采样与符合全局优化约束项所需的拓扑提取。并先将处理后的 STEP 格式的文件存储为含有顶点位置、32 个边采样点位置， 32×32 个面采样点位置的以及基于 STEP 文件给出的边面邻接关系的 NPZ 文件。

4.2 实验环境搭建

实验环境运行在 Docker 内部，以 NVDnet [8] 为基础镜像创建了实验环境。

4.3 界面分析与使用说明

代码通过传入处理后的模型文件路径来进行全局优化计算，计算所得的文件将以 `extraction.npz` 为后缀存储在模型文件夹下，运行文件命令如下

```
1 python GlobOptimize.py --model_folders /path/to/model/folders
```

其中因 Gurobi [3] 软件若需要在不同机器上运行必须使用运行证书，因此需要先申请证书并手动修改 `GlobOptimize.py` 中 `option` 所含的证书信息以正确运行。

4.4 创新点

- 源代码的输入仅可以是原论文中 ComplexNet 的预测输入，现可扩展至处理后的 ABC [5] 数据集上数据，扩展了其应用范围。
- 对 ABC 的数据集进行了一定程度的处理，使其在其中的图形拓扑结构能够成为一个合法的流形。
- 简化了源代码的代码逻辑，源代码过于冗余，长达 2000 余行，简化后代码量约为原先的 1/10。

5 实验结果分析

如图4所示，全局优化方法能够在一定程度上剔除错误的图元。然而应用在真实数据集上的情况并不理想，在一定的情况下难以成功得到正确结构，然而输出的结果符合约束，如图5所示，输出会把两条连接的边的“有效性”置为 0，从而提升优化函数的目标值。

原因在 ABC 数据集不存在“有效性”这一重要。而对于真实的模型，因为考虑到顶点与边的数据相较于面更加脆弱，因此人为把所有的点、边置为了 0.5，面的有效性为 1.0，因此优化结果出现了错误。如果将原数据的顶点、边、与面的有效性都人为置为 1.0，则还是可以得到正确的拓扑结构。

此外尝试过增加约束条件，考虑到 ACBC 数据集内图形的特殊约束，即数据集内的顶点都是由三个面相交而来，则有约束项：

$$\sum_i \mathbf{FV}[i, j] = 3\mathbf{V}[j], \quad \forall i \in [N_f] \quad (6)$$

在增加该约束的情况下依旧无法成功获得正确优化后结果，其原因依旧在图元有效性问题上，若是手动设置图元有效性为 0.5，则因图元权重 $w_x = 10$ 而拓扑权重 $w_x = 1$ 而使得大量有效图元的有效性被设置为错误的 0，而保留错误的拓扑结构，但优化目标值却确实朝着变大的方向迭代。而考虑到本文本身目标为将优化操作迁移至 ABC 数据集并获得符合链复形约束的图形结构，以及数据构建过程中拓扑结构因 ABC 数据集本身的数据特性必须完全参照图元本身，则如果不将图元有效性设为 0.5 便会导致优化数据的失真。

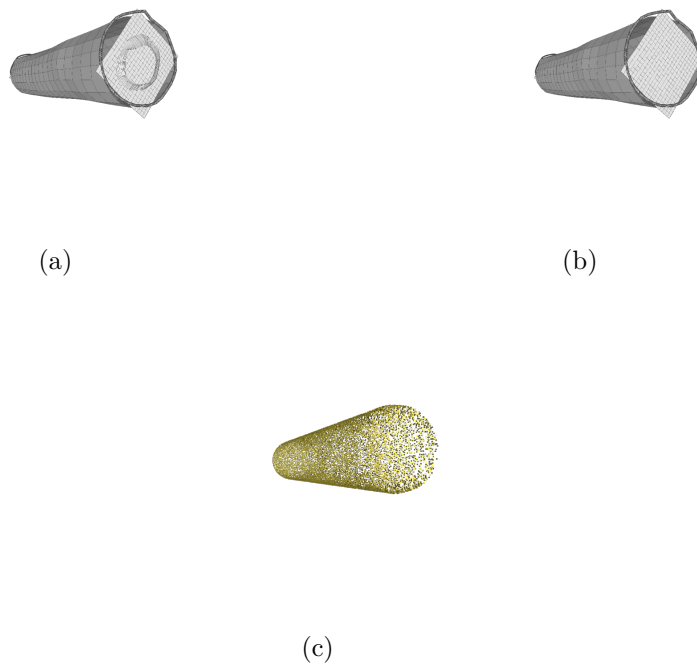


图 4. (a) 预测模型 (b) 优化后 (c) 原模型点云



图 5. 优化后获得错误拓扑结构

6 总结与展望

本文档基本实现了 ComplexGen 复形提取部分的主要功能，对模型预测结果进行拓扑全局优化（复形链提取），此外还将 ComplexGen 拓扑优化部分迁移至与论文不同的数据集上运行并尝试得到优化后正确的模型拓扑结构，然而结果存在一定的问题，希望在将来能够修复因缺失“有效性”这一输入而导致的全局优化错误的问题，使得模型优化仅依靠链复形自身存在的约束结构，从而得到更可靠的优化结果。

参考文献

- [1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.
- [2] David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. Variational shape approximation. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, page 905–914, New York, NY, USA, 2004. Association for Computing Machinery.
- [3] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2024.
- [4] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.
- [5] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [6] Lingxiao Li, Minhyuk Sung, Anastasia Dubrovina, Li Yi, and Leonidas J Guibas. Supervised fitting of geometric primitives to 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2652–2660, 2019.
- [7] Yangyan Li, Xiaokun Wu, Yiorgos Chrysathou, Andrei Sharf, Daniel Cohen-Or, and Niloy J. Mitra. Globfit: consistently fitting primitives by discovering global relations. *ACM Trans. Graph.*, 30(4), July 2011.
- [8] Yilin Liu, Jiale Chen, Shanshan Pan, Daniel Cohen-Or, Hao Zhang, and Hui Huang. Split-and-Fit: Learning B-Reps via structure-aware Voronoi partitioning. *ACM Transactions on Graphics (Proc. of SIGGRAPH)*, 43(4):108:1–108:13, 2024.
- [9] Yujia Liu, Stefano D’Aronco, Konrad Schindler, and Jan Dirk Wegner. Pc2wf: 3d wire-frame reconstruction from raw point clouds. *arXiv preprint arXiv:2103.02766*, 2021.

- [10] Liangliang Nan and Peter Wonka. Polyfit: Polygonal surface reconstruction from point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2353–2361, 2017.
- [11] Charlie Nash, Yaroslav Ganin, S. M. Ali Eslami, and Peter W. Battaglia. Polygen: an autoregressive generative model of 3d meshes. In *Proceedings of the 37th International Conference on Machine Learning*, ICML’20. JMLR.org, 2020.
- [12] Ruwen Schnabel, Patrick Degener, and Reinhard Klein. Completion and reconstruction with primitive shapes. *Computer Graphics Forum*, 28(2):503–512, 2009.
- [13] Gopal Sharma, Difan Liu, Evangelos Kalogerakis, Subhransu Maji, Siddhartha Chaudhuri, and Radomír Měch. Parsenet: A parametric surface fitting network for 3d point clouds, 2020.