

# 浙江大学

## 读书报告



题目 MeshCNN：一种先进的网络

姓名 周杰辉

学号 21951103

指导教师 李启雷

专业班级 软件工程 1905

所在学院 软件学院

日期 2019 年 12 月

## 摘要

多边形网格可有效表示 3D 形状。它们显式地捕获形状表面和拓扑，并利用非均匀性来表示较大的平坦区域以及尖锐、复杂的特征。但是，这种不均匀性和不规则性限制了使用结合卷积和池化操作的神经网络进行网格分析的尝试。在本文中，我们利用 MeshCNN（专门为三角形网格设计的卷积神经网络）直接利用网格的独特属性来分析 3D 形状。与经典的 CNN 相似，MeshCNN 通过利用固有的测地线连接，将在网格边上运行的特殊卷积和池化层相结合。在边及其伴随三角形的四个边上应用卷积，并通过保留表面拓扑的边收缩操作应用池化，从而为后续的卷积生成新的网络连接。MeshCNN 学习哪些边会收缩，从而形成任务驱动的过程，网络将在其中暴露和扩展重要的特征，同时丢弃冗余特征。我们展示了 MeshCNN 在应用于 3D 网格的各种学习任务上的有效性。

**关键字：**卷积神经网络、形状分析、几何深度学习、形状分割

## Abstract

Polygonal meshes provide an efficient representation for 3D shapes. They explicitly capture both shape surface and topology, and leverage non-uniformity to represent large flat regions as well as sharp, intricate features. This non-uniformity and irregularity, however, inhibits mesh analysis efforts using neural networks that combine convolution and pooling operations. In this paper, we utilize the unique properties of the mesh for a direct analysis of 3D shapes using MeshCNN, a convolutional neural network designed specifically for triangular meshes.

Analogous to classic CNNs, MeshCNN combines specialized convolution and pooling layers that operate on the mesh edges, by leveraging their intrinsic geodesic connections. Convolutions are applied via an edge collapse operation that retains surface topology, thereby, generating new mesh connectivity for the subsequent convolutions. MeshCNN learns which edges to collapse, thus forming a task-driven process where the network exposes and expands the important features while discarding the redundant ones. We demonstrate the effectiveness of MeshCNN on various learning tasks applied to 3D meshes.

**Key words:** Convolutional Neural Networks, Shape Analysis, Geometric Deep Learning, Shape Segmentation

# 目 录

第 1 章 引言 .....	1
第 2 章 相关工作 .....	5
2.1 多视图的 2D 投影 .....	5
2.2 体积 .....	5
2.3 图结构 .....	6
2.4 流形 .....	6
2.5 点云 .....	7
第 3 章 在网格上应用 CNN .....	9
3.1 不变卷积 .....	9
3.2 输入特征 .....	10
3.3 全局顺序 .....	10
3.4 池化 .....	11
第 4 章 方法 .....	12
4.1 网格卷积 .....	12
4.2 网格池化 .....	13
4.3 网格上池化 .....	14
第 5 章 实验 .....	16
5.1 数据处理 .....	16
5.1.1 增强 .....	16
5.2 网格分类 .....	16
5.2.1 SHREC 数据集 .....	17
5.2.2 立方体雕刻 .....	18
5.3 网格分割 .....	20
5.3.1 COSEG 数据集 .....	21
5.3.2 人体分割数据集 .....	22
5.4 额外评估 .....	24
5.4.1 计算时间 .....	24
5.4.2 细分鲁棒性 .....	25
5.4.3 不变特征 .....	26
第 6 章 讨论与未来工作 .....	27
6.1 不变卷积 .....	27
6.2 空间自适应池化 .....	27

6.3 局限性和未来工作 .....	28
参考文献 .....	29
附录 .....	35
神经网络训练配置 .....	35

## 第 1 章 引言

三维形状在计算机图形学领域十分普遍，在计算机视觉和计算几何学中也是主要的研究对象。我们周围的形状，尤其是描述自然实体的形状，通常由连续的表面组成。由于计算的原因，并且为了便于数据处理，对于 3D 形状的各种离散近似被用来表示一系列应用程序中的形状。多边形网格表示（简称网格）是许多人的首选，它可以用过 3D 空间中的一组 2D 多边形来近似曲面。网格提供了有效、不均匀的形状表示。一方面，只需要少量的多边形即可捕获大而简单的表面。另一方面，表示灵活性可在需要时支持更高的分辨率，从而可以忠实地重建或描绘通常在几何上复杂的显著形状特征。网格的另一个优势是具有对连接信息进行编码的固有能力。这形成了表面的综合表示。

与另一个流行的选择（点云）相比，这些优势显而易见。尽管点云简单，并且与常见的数据采集技术（扫描）有直接的关系，但当需要更高质量和保留尖锐的形状特征时，点云表示就力有未逮了。

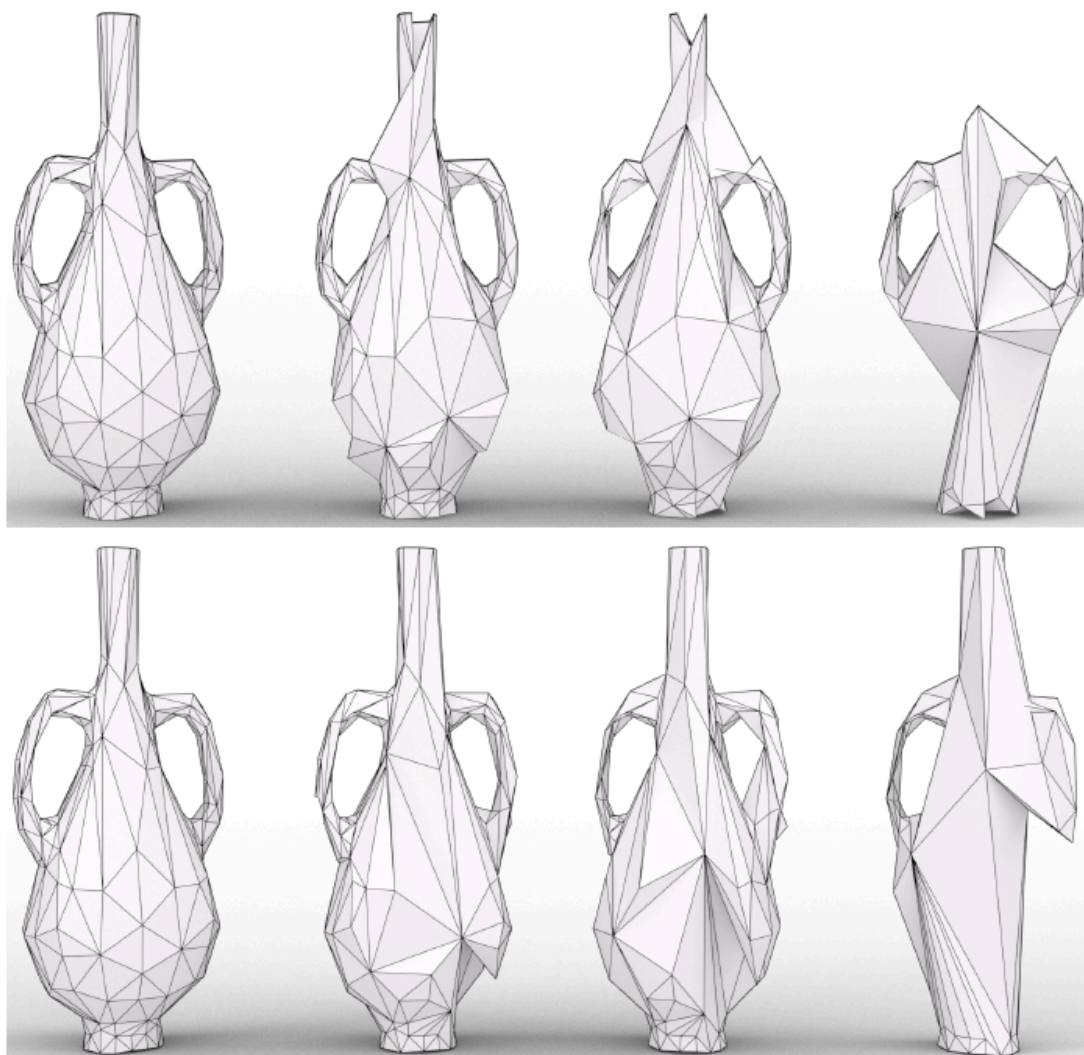


图 1 网格池化在不规则结构上运行，并在空间上适应任务。与几何简化（删除具有最小几何变形的边）不同，网格池化将哪些边收缩的选择交给网络。上行：MeshCNN 训练后可对花瓶是否具有手柄进行分类，下行：MeshCNN 训练的是花瓶是否具有瓶颈。

近年来，在图像上使用卷积神经网络（CNN）已显示出在各种任务（例如分类和语义分割）上的出色表现。成功的秘诀在于卷积、非线性和池化层的结合，从而形成了一个框架，该框架对于输入中无关变量的波动是鲁棒的。但是，由于图像是在规则的离散值网格中表示的，因此扩展 CNN 以在不规则结构上工作并不容易。

最初的方法是通过使用规则表示来使 CNN 适应不规则数据：将 3D 形状映射到多个 2D 投影或 3D 体素网格。虽然这些方法可以直接使用众所周知的图像 CNN 运算符，但它们的间接表示却需要浪费大量多余的 CNN 计算（例如，在未占用的体素上进行卷积）内存。

更有效的方法将 CNN 直接应用于不规则和稀疏的点云表示。尽管这些方法得益于紧凑的输入表示形式，但它们本质上是不考虑局部表面的。此外，邻居和连通性的概念定义不清，使得卷积和池化操作的应用变的不平凡。这种含糊不清导致了一系列旨在克服这一挑战的工作。

为了挖掘原始网格表示的自然潜力，我们提出了 MeshCNN：类似于众所周知的 CNN 的神经网络，但专门为网格设计。MeshCNN 直接在不规则的三角形网格上运行，执行与独特的网格属性协调设计的卷积和池化操作。在 MeshCNN 中，网格的边缘类似于图像中的像素，因为它们是所有操作的基本构件。我们选择使用边，因为每条边都恰好伴随两个面（三角形），这定义了四个边的自然固定大小的卷积邻域（参见图 2）。我们利用一致的面法线顺序应用对称卷积运算，该运算可以学习旋转、缩放和平移不变的边缘特征。

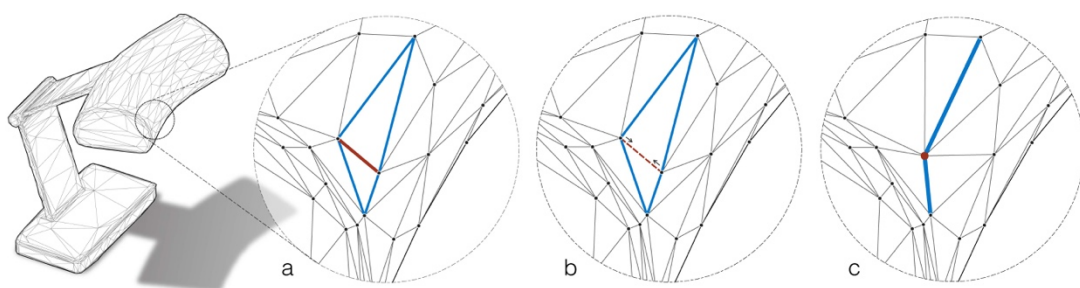


图 2 (a)通过对边(红色)的两个伴随三角形的四个边(蓝色)组成的邻居进行卷积，在边上计算卷积。(b) 和 (c) 中显示了池化步骤。

MeshCNN 提出了网格池化，该池化可在不规则结构上运行并在空间上适应任务。在 CNN 中，池化会对网络中特征的数量进行下采样，从而学习消除信息量较少的功能。由于特征在边上，向下采样的一种直观方法是使用众所周知的网格简化技术——边收缩。传统的边收缩旨在通过去除引入最小几何形变的边来保持原始形状。尽管可以使用传统的边收缩进行池化，但我们选择了学习型边收缩：将哪些边收缩的选择权交给网络。删除的边是其特征对使用的目标贡献最小的边（参见图 1 和图 8）

为了增加灵活性并支持各种可用数据，每个池化层将网格简化为预定的恒定边数。虽然我们的方法在数学上不依赖于相似变换和边排序，但不能保证它对于表面上的不同三角剖分或顶点移动是不变的。为了概括不同的网格划分和顶点变



化，我们在训练中采用了多种数据增强技术，例如边翻转和顶点扰动，这在实践中保证了鲁棒性。为了说明我们方法的适用性，我们对形状分类和分割任务进行了各种实验，并在通用数据集和高度非均匀网格上展示了优于最新方法的结果。

## 第 2 章 相关工作

我们在工作中使用的许多算子都是基于经典的网格处理技术，或更具体地说，是网格简化技术。特别地，我们将边收缩技术用于任务驱动的池化算子。传统的网格简化技术旨在以最小的几何形变减少网格元素的数量，在这项工作中，我们使用网格简化技术来降低神经网络环境下特征图的分辨率。接下来，我们将根据输入表示类型重新组织有关使用神经网络进行 3D 数据分析的相关工作。

### 2.1 多视图的 2D 投影

通过从各种角度用 3D 形状的 2D 投影表示 3D 形状，可以充分利用 2D 域中的现有技术和体系结构。这些已渲染的图像集被标准 CNN 模型用作进行后续处理的输入。Su 等人是第一个将多视图 CNN 用于形状分类任务，但是，这种方法（按原样）无法执行语义分割。后来，Kalogerakis 等人提出了一种更全面的形状分割多视图框架：为每个视图生成图像级分割图，然后使用 CRF(经过训练的端到端)解决标签的一致性。Qi 等探索了基于视图的方法和基于体积的方法，并注意到了第一种方法与当时可用方法的优越性。最近，一些工作考虑了每个视图的多层表示。Sarkar 等使用多层高度图以及新颖的多视图合并技术表示 3D 形状。Zhou 等在运动剪辑上训练以执行视图推断。Gomez-Donoso 等使用 3D 点云的三个切片（每个轴一个）进行物体分类。

### 2.2 体积

将 3D 形状转换为二进制体素形式可以提供类似于图像的 2D 网格的基于网格的表示。可以以直接的方式将应用于 2D 网格的操作扩展到 3D 网格，从而允许将基于图像的常见方法自然地转移到形状域。Wu 等率先提出了这一概念，并提出了一种处理体素化形状的 CNN 以进行分类和实现。Brock 等使用基于体素的变分自动编码器解决了形状重构问题，Tchapmi 等将三线性插值和条件随机场（CRF）与体积网络相结合，以促进语义形状分割。Hanocka 等使用体积形状表示来训练网络以回归基于网格的扭曲场以进行形状对齐，并将估计的变形应用于原始网格。

尽管它具有诱人的简单性，但其体积表示仍需要大量计算，这需要占用大量内存。为了缓解这种情况，一种加速策略已经被提出来了，其中利用空间中形状占比的稀疏性来压缩表示。

## 2.3 图结构

图结构是允许不规则性的基于网格的表示形式的泛化。为了支持基于图的数据分析，相当大的注意力已集中到将神经网络应用到涉及以图结构表示的数据（主要是社交网络、通信中的传感器网络或基因数据）的流形任务上。一种方法主张处理图表示的拉普拉斯算子，因此在频域上工作。另一种方法是通过提取局部连通区域并将其转换为要由神经网络处理的规范形式来直接处理图。Atwood 等提出了扩散卷积，其中在每个节点上应用扩散来确定其局部领域。Monti 等使用图空间域将表面参数化为局部小块。Xu 等使用表面小块上的定向卷积来完成语义分割任务。Yi 等在 3D 分割任务中在频域中使用图卷积。Kostrikov 等使用拉普拉斯曲面网络，用于开发 3D 形状的生成模型。Such 等引入了图上顶点滤波的概念，但没有包含用于特征聚合的池化操作。这些方法通常在图中的顶点上运行。

## 2.4 流形

Masci 等人的开拓性工作引入了深度学习网格上的局部特征（类似于 Kokkinos 等人的本征网格描述符），该算法用于对应和检索。具体来说，他们演示了如何使卷积运算成为网格所固有的。

通常，流形上的局部小块近似于欧几里得（空间中的局部小块）。通过将 3D 流形参数化为 2D，可以使用标准 CNN 将此特征用于流形形状分析。Boscaini 等使用顶点频率分析来学习局部 3D 形状描述符。另一种方法将球体形状参数化为平面上的平坦环面，其中明确定义了卷积。最近，Jiang 等将 3D 网格参数化为二十面体球，并提出了一种新的高效卷积算子。Poulenard 等定义了一个新的卷积层，允许在整个网络层中传播测地信息。

Verma 等提出了一种图神经网络，其中卷积运算的每个顶点的邻域不是预先定义的，而是根据其特征动态计算的。Tatarchenko 等引入了切线卷积，其中每个点周围的一个小邻域用于重构应用卷积的局部函数。与此前的工作不同，他们通过在规则 3D 网格上进行下采样来合并池化操作。一些生成模型也被提了出来。Litany 等引入了形状完成的自动编码器。Ranjan 等演示了网格自动编码器如何生成 3D 面部。

参见 Bronstein 等对几何深度学习的全面调研。与之前的方法相比，我们的方法的独特之处在于我们的网络操作是专门为适应网格结构而设计的。特别地，我们学习了一个独特的池化算子，它可以根据目标任务选择要简化的区域。

据我们所知，这是第一个提出的工作 (i) 在网格边缘上的等变卷积，以及 (ii) 适应于当前任务的学习到的网格池化操作。Ranjan 等人已经提出了针对网格自动编码器的固定池化操作。在图神经网络的背景下，Ying 等人提出了学习型的池化操作。Monti 等利用对偶图卷积模型提出了一种提取边特征的卷积。然而，由于这些方法适用于一般图形，因此它们并未利用独特的网格属性，及多边形网格面。在这项工作中，我们定义了网格的算子，这些算子利用了它们的独特结构和属性。这使我们能够定义等变卷积和相似不变输入特征。

## 2.5 点云

点云表示形式可以说是所有表示形式中最简单的，它提供了基本 3D 形状的近似。与数据采集的紧密关系以及从其他表示形式转换的便捷性，使点云成为数据分析的经典候选者。因此，最近的努力集中在开发使用神经网络进行点云形状分析的技术。PointNet 建议使用  $1 \times 1$  的卷积，然后用全局最大池化使得阶数不变形。在其后续工作中，PointNet++ 对点进行分区以更好地捕获局部结构。Wang 等考虑到局部点邻域信息，并基于特征空间中的距离执行由点之间的相似性驱动的动态更新。尽管大多数基于点的方法都侧重于全局或中级属性，Guerrero 等提出了一个网络来估计局部形状属性，例如，来自原始点云的法线和曲率，而 Williams 等从点云中学习曲面重建的几何先验。Atzmon 等通过将点云函数映射到体积函数，在点云上定义了高效的卷积算子。这使得该方法对于点的顺序不变，并且对数据中的某些形变具有鲁棒性。最近，Li 等人提出了

PointCNN，将卷积的概念从局部网络扩展到了位于其欧几里得邻域上的点的 $\chi$ -卷积。

在这项工作中，与以前的工作不同，我们依靠网格的边来提供非均匀的测地邻域信息，并具有一致数量的卷积邻居。不变特征计算是在边上执行的，同时以符合几何形状和拓扑的方式利用网格抽取技术（例如边收缩）。

## 第 3 章 在网格上应用 CNN

计算机图形学中最基本,最常用的 3D 数据表示法是非均匀多边形网格。大的平坦区域使用少量的大多边形,而细节区域使用大量的多边形。网格明确表示表面的拓扑: 忠实地描述复杂的结构,同时消除与附近表面的接近性(见图 2-1)。

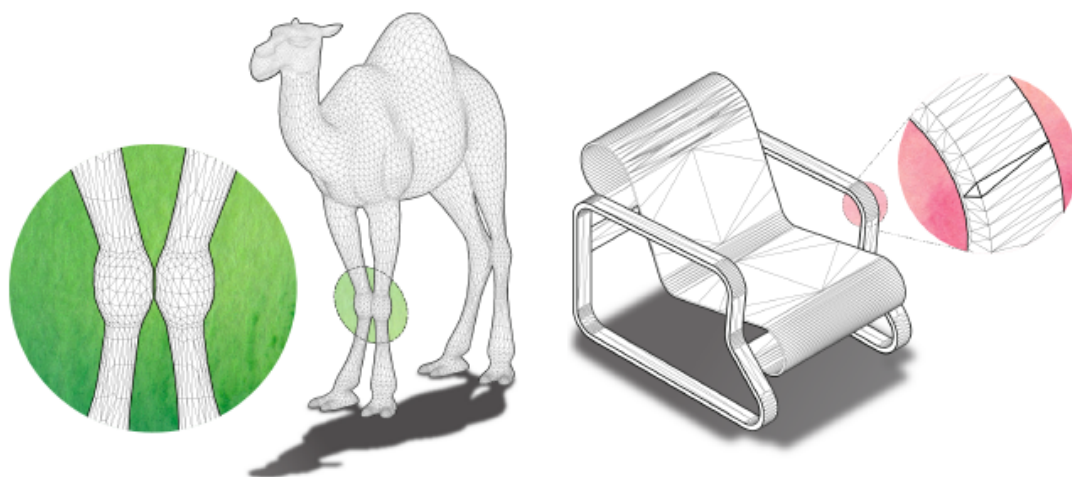


图 3 多边形网格的优点。左: 形状结构的准确描绘。网格与点云不同, 尽管它们位于欧几里得空间中, 但可以通过测地间隔轻松的传达骆驼关节(放大)的不同特性。右: 自适应非均匀表示。较大的平坦区域可以由少量的大多边形表示, 而详细区域可以由大量的小多边形表示。

将 CNN 范式直接应用到三角形网格, 需要对 CNN 的标准构建块进行类似的定义和实现: 卷积和池化层。与在离散值的规则网格上表示的图像相反, 网格分析的关键挑战是网格固有的不规则形和不均匀性。在这项工作中, 我们旨在利用这些具有挑战性的独特属性, 而不是绕开它们。因此, 我们将网络有意设计为将卷积和池化操作直接应用于网络元素, 并避免转换为规则且统一的表示形式。

### 3.1 不变卷积

在我们的设置中, 假定所有形状都表示为流行网格, 可能带有边界。这样的假设保证了每条边最多伴随两个面(三角形), 因此与两个或四个其他的边相邻。面的顶点按逆时针方向排序, 从而定义了每条边的四个邻居的两种可能的排序方式。例如, 参见图 4, 其中  $e$  的 1 环邻居可以按如下顺序排序:  $(a, b, c, d)$  或

(c, d, a, b)，具体取决于哪个面定义了第一个邻居。这使得卷积接收场令人费解，阻碍了不变特征的形成。

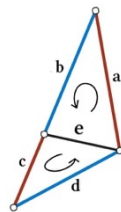
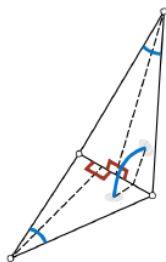


图 4 示意图

我们采取了两种措施来解决此问题，并保证网络中相似性转换（旋转、平移和缩放）的不变形。首先，我们精心设计了一条边的输入描述符，使其仅包含相对于相似变换内不变的相对几何特征。其次，我们将四个 1 环边聚合为两对具有模糊性的边（例如 a 和 c，以及 b 和 d），并通过在每对边缘上应用简单的对称函数（例如  $\text{sum}(a, c)$ ）。将卷积应用于新的对称特征，从而消除了任何顺序歧义。

### 3.2 输入特征

输入边特征时每条边的 5 维向量：二面角，两个内角和两个面的边长度比。边比率是边的长度和每个相邻面的垂直（虚线）之比。我们对两个基于面的特征（内角和边长比）中的每一个进行排序，从而解决了顺序的歧义并保证了不变性。观察到这些特征都是相对的，从而使它们对于平移，旋转和均匀放缩具有不变性。



### 3.3 全局顺序

边的全局顺序是特定形状的边缘数据（输入特征）进入网络的顺序。由于在局部邻域内进行卷积，因此该顺序在卷积阶段没有影响。通过扩展，完全卷积的

任务（例如分段）不受其影响。对于需要全局特征聚合的任务（例如分类），我们遵循 Qi 等人的建议。在 PointNet 中，放置一个连接网络卷积部分和全连接部分之间的全局平均池化层。该层使初始顺序无关紧要，从而保证了转换的不变性。

### 3.4 池化

网格池化是通过边收缩来完成的，如图 2(b) 和 (c) 所示，虚线边收缩到一个点，然后四个伴随的边（蓝色）合并到 (c) 中的两个（蓝色）边。请注意，在此边收缩操作中，五条边被转换为两条边。（最小范数）边特征为算子确定了优先级，从而使网络可以选择要简化的网格部分以及要保留的网格部分。这将创建一个任务感知过程，网络将学会根据任务确定对象部分的重要性（见图 1）。

我们的简化性质的一个显著优势是，就池化层的输出尺寸而言，它在到达最终的全连接层之前就提供了灵活性。池化还有助于增强初始网格三角剖分的鲁棒性。尽管它不能为三角剖分提供同变性，但实际上，通过不断收缩边和简化网格，尽管初始曲面细分有所不同，但我们仍观察到收敛到相似表示的情况。



## 第 4 章 方法

基于网格的（例如，图像）表示可以方便地在单个矩阵中同时提供空间邻居（连接性）和特征。但是，由于不规则网格不符合此格式，因此我们必须与连接分开定义特征。我们通过在网格的标准构造中工作来完成此任务。

网格由对  $(V, F)$  定义，其中  $V = \{v_1, v_2 \dots\}$  是  $\mathbb{R}^3$  中顶点位置的几何，而  $F$  定义连通性（三角网格的顶点三元组）。给定  $(V, F)$ ，还使用边  $E$ （一组顶点对）定义网络连接。

所有网格元素  $V, F$  和  $E$  都可以与各种特征（例如法线或颜色）关联。在这项工作中， $E$  还具有一组特征。边特征从一组相似的不变几何特征开始（在图像的情况下相当于 RGB 值），并随着它们通过网络层而发展出更高的抽象度。

在我们的设置中，网格为网络提供了两个属性：卷积邻居的连接性和初始几何输入特征。一旦提取了输入特征，网格顶点就没有任何意义。边收缩操作之后的新顶点位置对分类和分割任务没有影响，并且仅出于可视化目的对其进行计算。

接下来，我们扩展并提供有关网格卷积，网格池化和网格上池化操作的详细信息。

### 4.1 网格卷积

我们为边定义卷积算子，其中使用四个伴随邻居定义空间支持（图 3）。回想卷积是核  $k$  与邻域之间的点积，因此边特征  $e$  及其四个相邻边的卷积为：

$$e \cdot k_0 + \sum_{j=1}^4 k_j \cdot e^j \quad (1)$$

其中  $e^j$  是  $e$  的第  $j$  个卷积邻居的特征。请注意，如图 4 所示， $e$  的四个邻居，即  $(e^1, e^2, e^3, e^4)$  是  $(a, b, c, d)$  或  $(c, d, a, b)$ ，这样每个滤波器值最多在两个可能的边上进行操作（例如  $a$  或  $c$  上的  $k_1$ ）。为了保证对输入数据的顺序具有卷积不变性，我们将一组简单的对称函数应用于不明确的对。这将生成一组新的卷积邻域，这些卷积邻域保证是顺序不变的。在我们的设置中，边  $e$  的接收场为

$$(e^1, e^2, e^3, e^4) = (|a - c|, a + c, |b - d|, b + d) \quad (2)$$

这将导致卷积操作，而该操作会忽略网格元素的初始顺序，因此无论什么顺序都会产生相同的输出。回想一下，可以使用通用矩阵乘法（GEMM）来实现多通道张量与核的卷积：通过将图像扩展（或展开）为列矩阵（即 Jia 等人提到的 *im2col*）。等效地，我们构建一个未展开的矩阵以有效地执行卷积运算。

实际上，我们可以通过将所有边特征聚合到  $(n_c \times n_e \times 5)$  的特征张量中使用高度优化的批处理算子（例如 *conv2D*），其中  $n_e$  是边数， $n_c$  是特征通道数，5 是指边和其卷积邻居（等式 2）。该矩阵乘以权重矩阵使用标准 GEMM 进行卷积。

卷积操作之后，将生成一个新的批特征张量，其中新的特征数量等于卷积核的数量（就像在图像中一样）。请注意，在每个池化阶段之后，新的连通性将为下一个卷积定义新的卷积邻居。

## 4.2 网格池化

通过确定三个概括了池化概念的核心操作，我们将传统的池化扩展到不规则数据：

- 1) 定义给定邻接的池化区域
- 2) 合并每个池化区域中的要素
- 3) 重新定义合并特征的邻接关系

对于在常规数据（例如图像）上进行合并，其固有地隐含了邻接关系，因此池化区域直接由所选内核大小确定。由于每个区域的特征（例如 *avg* 和 *max*）以产生另一个均匀间隔的网格的方式合并，因此再次固有地定义了新的邻接关系。提出了上面定义的三个常规池化操作后，很明显，传统池化是广义过程的特殊情况。

网格池化是广义池化的一种特殊情况，其邻接关系由拓扑决定。与图片不同，例如对于  $2 \times 2$  池化，其自然缩减因子为 4，我们将网格收缩定义为一列边收缩操作，其中每个这样的边收缩将五个边转换为两个边。因此我们可以通过添加

一个超参数来控制每个池化算子之后所需的网格分辨率，该超参数定义了池化后网格中目标边的数量。在运行期间，抽取网格邻接信息需要查询不断更新的特殊数据结构（有关详细信息，可以参见 Berg, 2008）。

我们通过边特征的大小对边收缩顺序（使用优先队列）进行优先排序，从而允许网络选择网格的哪些部分与解决任务有关。这使网络能够非均匀地收缩某些对损失最不重要的区域。回想一下，使两个面相邻的边收缩会导致三条边的删除（如图 2 所示），因为两个面都变成一条边。每个面包含三条边：最小边和最小边的两个邻居（请参见图 2 中红色的最小边和蓝色的相邻边）。通过获取每个特征通道的平均值，可以将每个面中三条边的每个特征合并为一个新的边特征。

根据边特征的强度优先考虑边收缩，将其视为  $\ell_2$  范数。如图 5 所示，这些特征被聚合，其中有两个合并操作，分别用于最小边特征  $e$  的每个伴随三角形，从而产生两个新的特征向量（表示为  $p$  和  $q$ ）。两个三角形的通道索引  $i$  中的边特征由下式给出

$$p_i = \text{avg}(a_i, b_i, e_i), \text{ and } q_i = \text{avg}(c_i, d_i, e_i) \quad (3)$$

边收缩后，将更新半边数据结构以用于后续的边收缩。

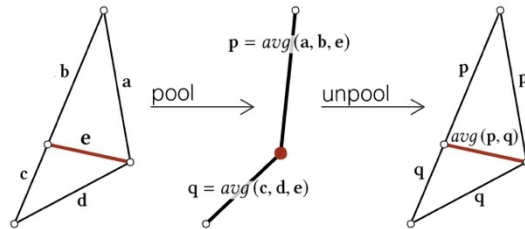


图 5 在网格池化和上池化中的特征聚合

最后，请注意，并非每个边都可以折叠。在我们的设置中，不允许产生非流形面的边收缩，因为它违反了四个卷积邻居的假设。因此，如果一条边在其 1-环的交点上具有三个顶点，或者它具有两个边界顶点，则认为该边对收缩无效（有关详细信息，请参见 Botsch, 2010）。

### 4.3 网格上池化

上池化是池化操作的（部分）逆过程。池化层降低特征激活的分辨率（编码或压缩信息），上池化提高特征激活的分辨率（解码或解压缩信息）。池化操作

记录来自合并操作的历史记录(例如,最大位置),并使用它们来扩展功能激活。因此,上池化没有可学习的参数,通常将其与卷积结合以恢复池化操作中丢失的原始分辨率。与卷积的结合有效地使上池化成为可学习的操作。

每个网格的上池化层都与池化层配对,以对网格拓扑和边特征进行上采样。上池化层通过在池化之前存储连接性来恢复上采样的拓扑(在网格池化之前)。请注意,对连接进行上采样是可逆的操作(就像在图像中一样)。在非池化边特征计算中,我们保留了一个图,该图存储了从原始边(池化之前)到新边(池化之后)的邻接关系。每个未池化的边特征是池化的边特征的加权组合。图 5 展示了平均上池化的情况。

---

## 第 5 章 实验

MeshCNN 是将 CNN 直接应用于三角形网格的一种通用方法，具有许多应用。使用第 4 章中描述的 MeshCNN 的构建块，我们可以构建不同的网络配置来解决不同的任务。向传统的 CNN 一样，这些构件提供了即插即用的框架。为了提高计算效率，在池化操作中，每个池化操作仅汇总一次特征。在顺序进行边排序和收缩的同时，这种松弛允许在 GPU 上执行特征聚合操作，从而提高了计算效率。

接下来，我们演示 MeshCNN 在分类和分割任务上的性能。有关使用的网络体系结构的详细信息，请参阅附录 A。

### 5.1 数据处理

在所有集合中，我们将网格简化为大致相同数量的边。请注意，如前所述，MeshCNN 不需要所有样本的边数相同。但是，类似于 CNN 中图像的初始大小调整。但是，类似于 CNN 中图像的初始大小调整，几何网格抽取有助于降低输入分辨率，并因此降低训练所需的网络容量。由于分类任务学习是全局形状描述，因此与分割任务（2250 条边）相比，我们通常使用较低的分辨率（750 条边）。

#### 5.1.1 增强

存在几种形式的数据扩充，用于为网络生成更多的数据样本。请注意，由于我们的输入特征是相似不变的，因此应用旋转、平移和各向同性缩放（ $x, y$  和  $z$  相同）不会生成新的输入特征。但是，我们可以通过对  $x, y$  和  $z$  的顶点位置应用各向异性缩放来生成新的特征。 $\langle S_x, S_y, S_z \rangle$ （每个都从正态分布中随机采样  $\mu = 1$  和  $\sigma = 0.1$ ），这将更改网络的输入特征。我们还通过将顶点移到其 1 环中的随机顶点，来讲顶点移动到网络表面上的不同位置。此外，我们通过执行随机边缘反转来增强每个对象的细分。由于灵活的输入分辨率，我们还可以在训练之前收缩少量的随机边。

### 5.2 网格分类

### 5.2.1 SHREC 数据集

我们从 SHREC 数据集中对 30 个类别进行了分类，每个类别有 20 个样本，我们遵循 Ezuz 等人的方法，16 和 10 分别是每个类别的训练样本数，我们在 200 个周期后停止训练。由于我们没有 Ezuz 的确切的划分，我们的结果是对 3 个随机生成的 16 和 10 集合进行平均的结果。表 1 展示了结果。为了进行比较，我们直接利用 Ezuz 的评估，他们与 SG(bag-of-feature representation), SN(volumetric CNN), GI(CNN on fixed geometry images), 最后是 GWCNN(learned geometry images)。我们方法的优势显而易见。我们在图 6 中可视化了此数据集的网格池化的一些示例。我们观察到网格池化以一致的语义方式表现(见图 11)。

表 1 SHREC 30 分类，分组 16 和 10 是训练分组，最多可训练 200 个周期

Classification SHREC			
Method	Split 16	Split 10	
MeshCNN	<b>98.6 %</b>	<b>91.0%</b>	
GWCNN	96.6%	90.3%	} [Ezuz et al. 2017]
GI	96.6%	88.6%	
SN	48.4%	52.7%	
SG	70.8%	62.6%	

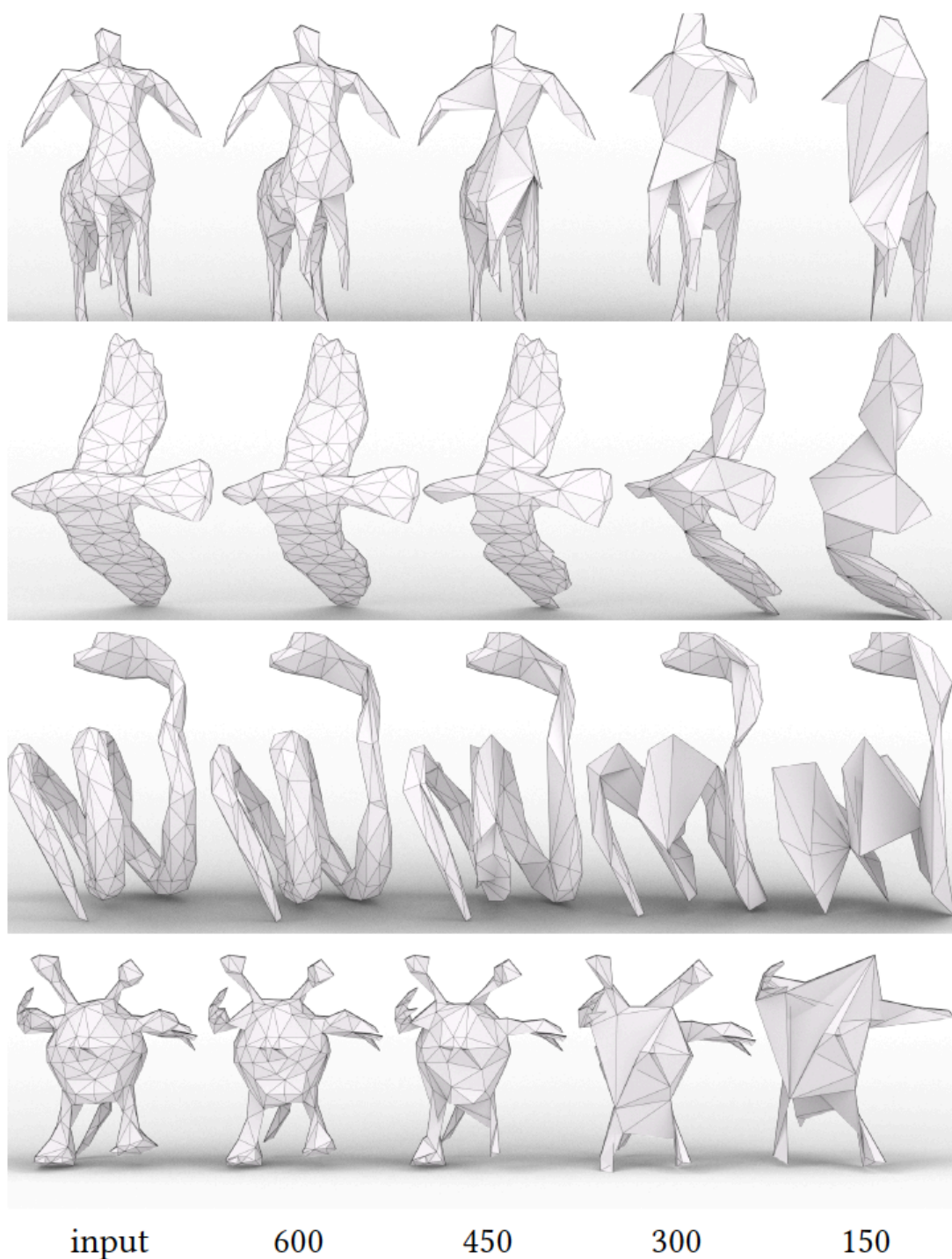


图 6 SHREC11 分类数据集的中间池化网络

### 5.2.2 立方体雕刻

为了说明 MeshCNN 的独特功能，我们对带有浅色图标雕刻的一组立方体进行了建模（请参见图 7）。我们使用 MPEG-7 二进制形状数据集中的 23 个



类，每个类大约有 20 个图标。我们在每个类别为测试机预流了三个图标，其余的用于训练。对于每个图标，我们随机采样 10 个不同的地方（位置，旋转和立方体面）以插入图标。每个立方体有大约 500 个面，这意味着详细的形状在平坦区域有更少的三角形，而不太详细的形状在平坦区域有更多的三角形。该集合包含总计 4600 个形状，其中训练/测试为 3910/690。我们计划在论文发表后，发布此数据集以及数据生成代码。

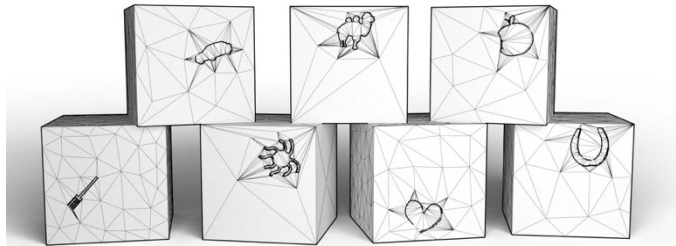


图 7 雕刻的立方体分类数据集。我们通过从 MPEG-7 中提取标签并将它们随机放置在立方体上来生成 23 种不同的类（例如，汽车、心脏、苹果等）。

我们训练 MeshCNN 对立方体进行分类。我们在表 2 中显示了定量结果。为了可视化网格池化对分类任务的影响，我们在每次网格池化操作后提取了中间结果（如图 8 所示）。可以观察 MeshCNN 如何学会减少与分类无关的边（平坦的立方体表面），同时保留图标雕刻内部和周围的边缘。

表 2 立方体雕刻结果

Cube Engraving Classification		
Method	Input Res	Test Acc
MeshCNN	750	<b>92.16%</b>
PointNet++	4096	64.26%
UGSCNN	750	61.97%

我们还在此集合上训练了基于点和基于网格的方法，并在表 2 中显示了结果。我们使用网格的测地采样训练了 PointNet++，并使用了输入特征的坐标和发现。此外，我们训练了一种基于球面网格 CNN 的方法 UGSCNN。我们将多维



数据集映射到输入的二十面体球形网格表示。尽管可以认为该例子是人为设计的，但这是要强调 MeshCNN 在 3D 形状上具有出色的表现，这些形状在几何分辨率上包含很大的差异。

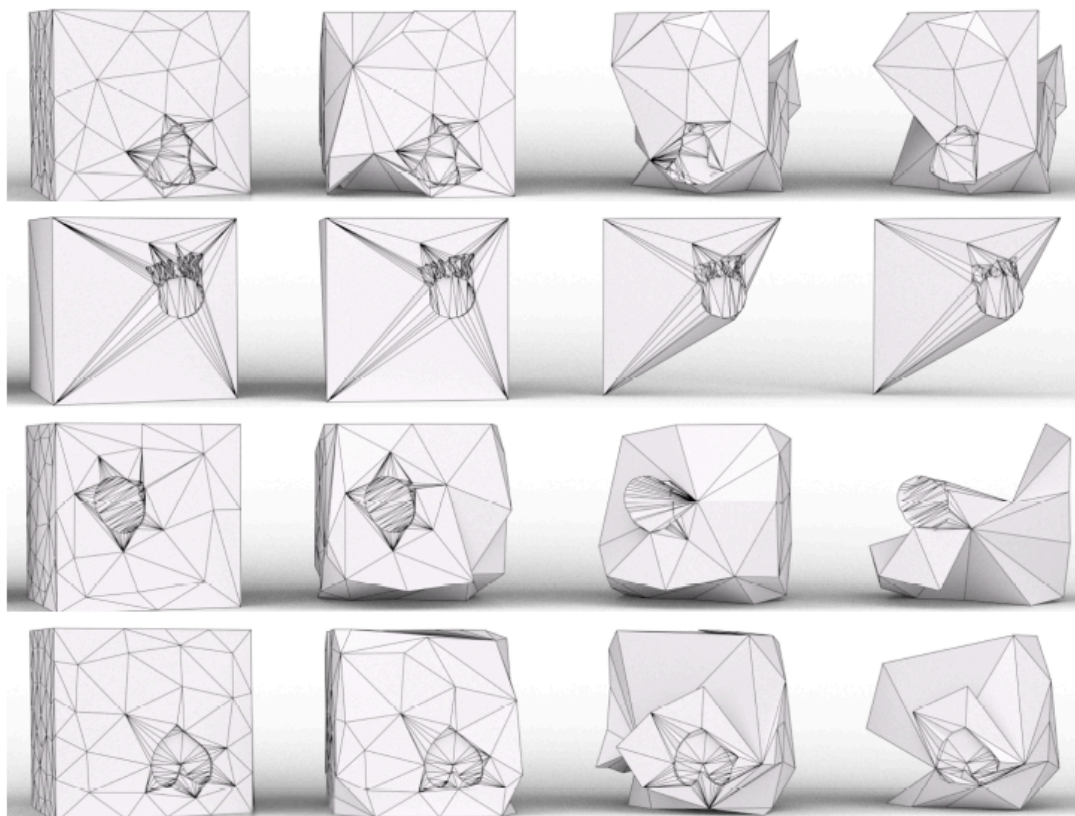


图 8 MeshCNN 经过训练可以预测图标雕刻的类别。观察网络如何学习保留重要边缘并删除有关分类任务的冗余边缘

### 5.3 网格分割

MeshCNN 的另一个应用是一致的形状分割，这是形状分析和合成中许多应用的重要组成部分。我们使用监督学习来训练 MeshCNN，在 COSEG 和 Human Body Segmentation 数据集上预测每条边属于特定段的概率。由于两个数据集都提供了每个面的真实标记，因此我们根据原始分辨率中的标签在简化的网格上生成了边语义标签。

最直接的 MeshCNN 语义分段配置将使用一些列网格卷积层（以及归一化和非线性激活单元）。但是，包含网格池化使 MeshCNN 能够学习由段驱动的边

收缩。回想一下，网格池化降低了输入网格的分辨率，该分辨率不再与真实边级别的标签一致。为此，我们使用网格上池化层将分辨率上采样回原始输入大小。

### 5.3.1 COSEG 数据集

我们在 COSEG 数据集的分割任务上评估了 MeshCNN 的性能，该数据集包含三个大集合：外星人，花瓶和椅子，每个分别包含 200、300 和 400 个模型。我们将每个形状类别划分为 85% / 15% 的训练/测试集。我们将 PointNet, PointNet++ 和 PointCNN 进行了比较，并在表 3 中报告了所有方法的最佳准确性。在该数据集上我们的技术取得了比所有其他方法更好的结果。

表 3 MeshCNN 对 COSEG 分割的评估

COSEG Segmentation			
Method	Vases	Chairs	Telealiens
MeshCNN ( <i>UNet</i> )	<b>97.27%</b>	<b>99.63%</b>	<b>97.56%</b>
MeshCNN ( <i>rand. pool</i> )	96.64%	99.23%	97.01%
PointNet	91.5%	70.2%	54.4%
PointNet++	94.7%	98.9%	79.1%
PointCNN	96.37%	99.31%	97.40%

我们认为这是由于我们的网络是针对网络结构量身定制的，这使它比其他策略更具优势。为了进一步证明这一点，我们还报告了随机池化（随机选取边进行收缩）情况下的结果，并表明这种变化会降低网络的性能。此外，图 9 显示了来自 MeshCNN 语义分段网络的最终分段预测，其中包含在测试集上的池化和上池化层。这也表明池化是如何使用目标问题的。

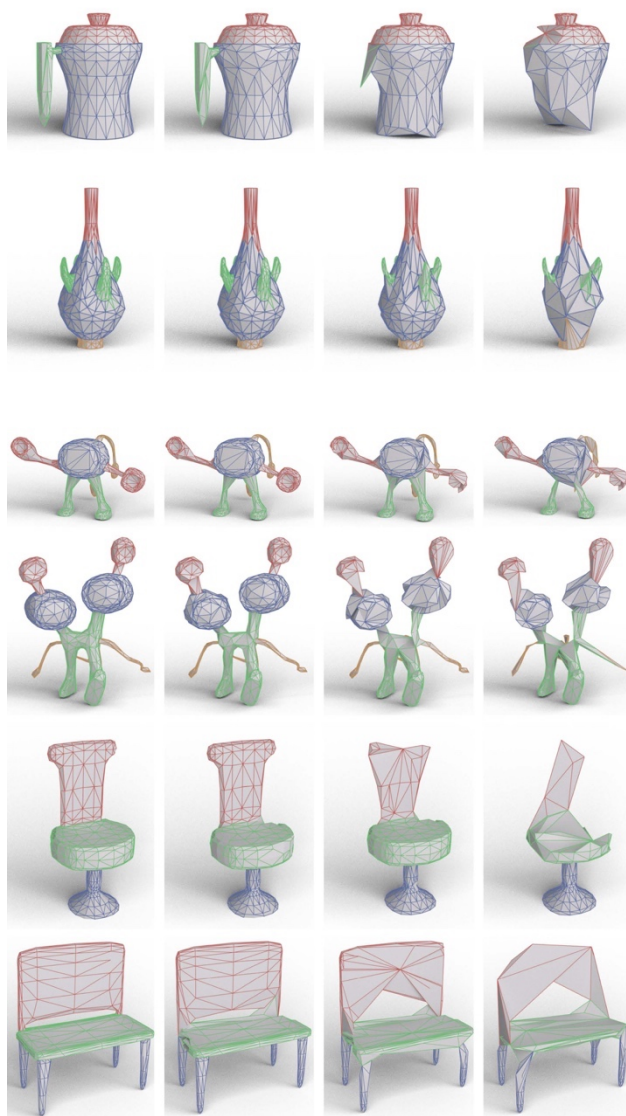


图 9 语义分割测试集结果。每个边的分段预测显示在左侧，然后是每个池化层之后的中间简化网格。出于可视化目的，中间网格中的边用最终的分割预测着色。请注意第一行：整个花瓶手柄已折叠到一个边上。

### 5.3.2 人体分割数据集

我们在 Maron 提出的人体分割数据集上评估了我们的方法。该数据集包含 SCAPE, FAUST, MIT, Adobe Fuse 的 370 个训练模型。测试集是 SHREC07 人体数据集的 18 种模型。根据 Kalogerakis 等人的论文，将模型手动分为八个标签。最近，Poulenard 和 Ovsjanikov 在该数据集上报告了其方法的结果，并于 GCNN, PointNet++ 和动态图 CNN 进行了比较。我们直接从他们的论文中获取报告结果，并将它们列出在表 4 中。我们在表中增加了 Haim 等人的最新结果。同样在这种情况下，MeshCNN 具有超过其他方法（一些基于图/流形而另一些基

于点) 的优势, 我们认为这是由于 MeshCNN 对网络结构和当前任务的适应性所导致的。图 10 给出了 MeshCNN 的一些定性结果。

表 4 人体分割结果对比

Human Body Segmentation			} [2018]
Method	# Features	Accuracy	
MeshCNN	5	<b>92.30%</b>	
SNGC	3	91.02%	
Toric Cover	26	88.00%	
PointNet++	3	90.77%	
DynGraphCNN	3	89.72%	
GCNN	64	86.40%	
MDGCNN	64	89.47%	

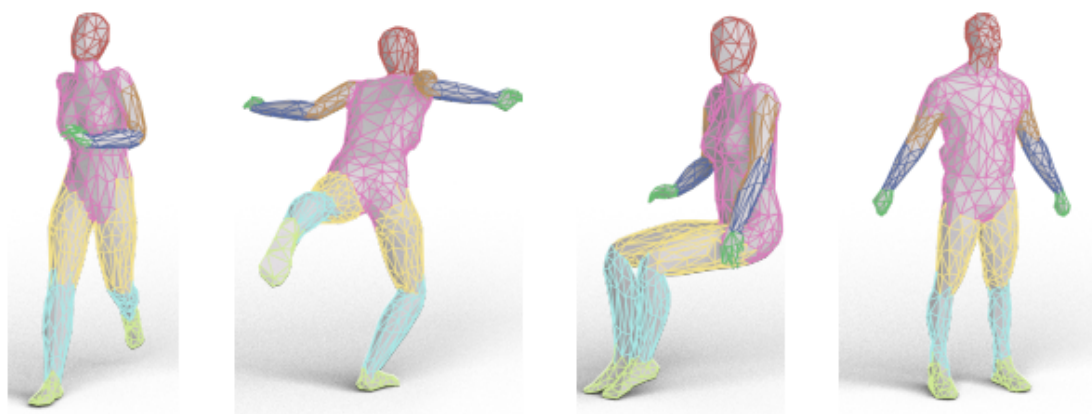


图 10 人体分割定性结果

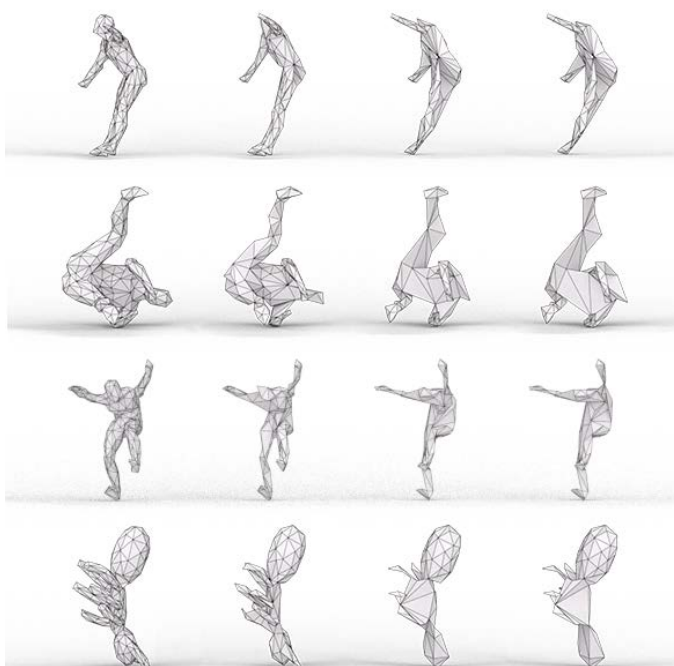


图 11 自适应池化有可能在类中公开语义，我们在形状分类（SHREC）任务的同一类中观察到一致的语义池化。例如，将人的头部类似地进行池化，而对于另一类（底部），相似的形状属性却被不同的汇集。

## 5.4 额外评估

### 5.4.1 计算时间

我们未经优化的 PyTorch 实现时，使用 GTX 1080 TI 显卡进行 2250/750 边分割/分类训练，每个示例平均花费 0.21/0.13 秒。



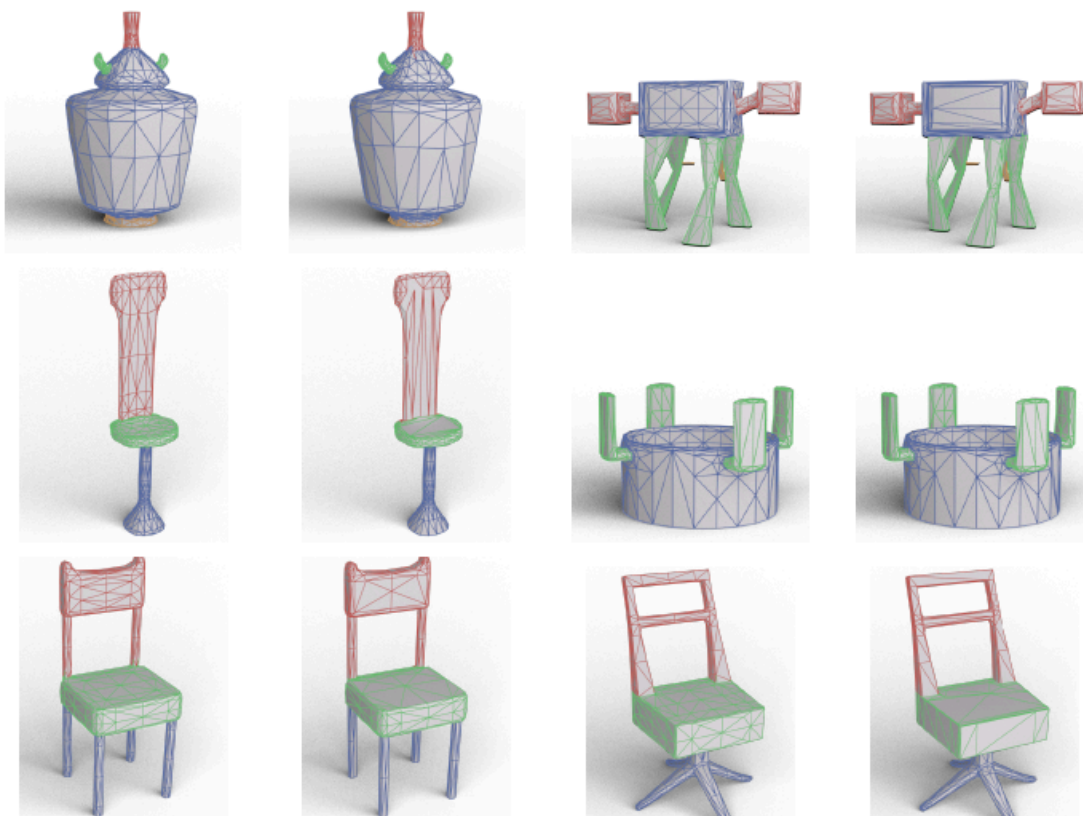


图 12 鲁棒性不同的三角剖分（语义分割）。左：测试形状，右：交替三角剖分。注意椅子子类中三角剖分的显著差异。

### 5.4.2 细分鲁棒性

我们通过使用 COSEG 分割数据集的一些定性和定量实验，检验了我们的方法对三角剖分差异的鲁棒性。为此，我们在原始（参考）数据集上进行训练，并比较三种不同集合的测试集准确性：参考测试集，重新网格化测试集和顶点扰动测试集。重新网格化的测试集是使用与参考集不同的网格划分程序生成的。顶点扰动集是通过沿其表面随机移动 30% 的顶点并向其 1 环中的随机顶点移动而生成的。定量和定性结果分别可见表 5 和图 12。这些结果表明网络已学会灵活应对这些类型的变化。请注意，参考集中的细分与重新网格化细分中的网络非常不同（请参见图 12 中的示例）。特别是，差异最大的组是椅子组（由于在平坦区域的重新网格化的方法不同）。确实，经过重新网格化的椅子组之间的这种差距导致表 5 中的椅子类下降了（10%）。

表 5 在两个鲁棒性集上的定量结果

Robustness			
Set	Vases	Chairs	Aliens
Reference	97.27%	99.63%	97.56%
Remeshing	96.33%	88.98%	97.77%
Perturbations	97.12%	99.60%	97.34%

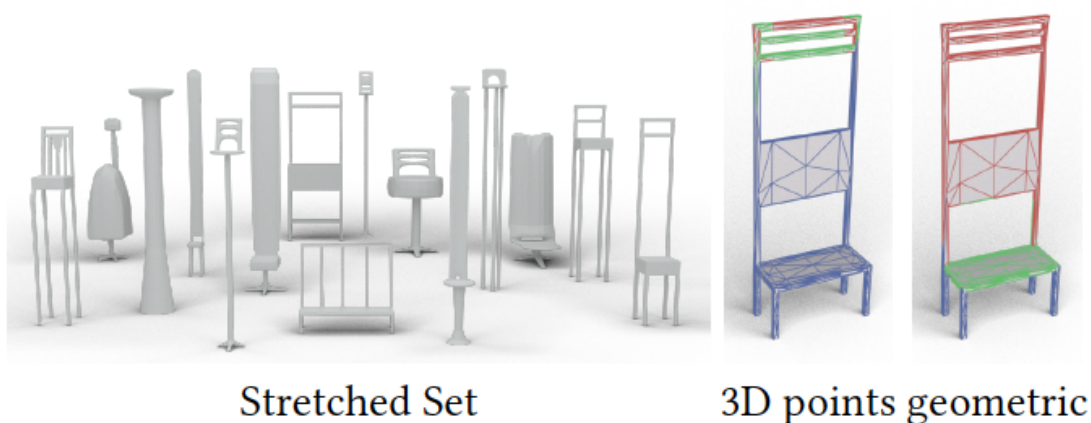


图 13 差异特征的鲁棒性。我们使用与标准 3D 点（边中点）相比具有几何特征的 MeshCNN 评估拉伸集（左）的通用性。尽管两者都在标准测试集上达到了高进度，丹玉 3D 点的情况（中）相比，MeshCNN 的几何特征（右）可以很好地推广到拉伸集。

### 5.4.3 不变特征

使用相对特征的一个显著优势是，MeshCNN 可以保证对旋转、平移和均匀缩放不变。本质上，常用的笛卡尔坐标对刚性变换敏感。为了说明这一点，我们在语义分割的任务上训练了 MeshCNN: (i) 使用不变的几何特征，(ii) 使用边中点  $(x, y, z)$  作为输入特征。为了评估学习的泛化性，我们沿垂直轴应用了非均匀缩放（没有对这些类别的增强进行训练）。我们的相对几何特征达到 98.44%，标准测试集为 99.63%，而绝对坐标退化为 78.27%，标准测试集为 99.11%。请注意，虽然我们的几何特征对于非均匀缩放不是不变的，但由于它们对位置不敏感，因此它们的泛化效果更好。

## 第 6 章 讨论与未来工作

我们介绍了 MeshCNN，这是一种直接在不规则三角形网格上直接使用神经网络的通用方法。我们的工作的关键贡献是针对不规则和非均匀结构量身定制的卷积和池化操作的定义和应用。这些操作有助于直接分析以原始形式表示为网格的形状，并因此受益于与具有不均匀结构的表面流形的表示相关的独特属性。

### 6.1 不变卷积

我们选择网格边作为网络运行的基本构建模块非常重要，因为边集规定了一种简单的方法来定义一个局部的、固定大小的邻域，用于在不规则结构上进行卷积。通过利用特定于三角形网络的独特对称性，我们消除了邻居顺序对偶性的歧义，以使变换具有不变性。我们通过选择输入边特征来补充这项工作，这些特征经过精心组织以仅包含相对几何属性而不是绝对位置。我们观察到，与使用绝对笛卡尔坐标相反，我们的微分特征不仅为相似度转换提供了不变性，而且还抑制了过度拟合。在不同对象之间执行语义相似池化的能力，进一步证明了网络的泛化性，自然可以带来更好的结果。因此，与普通表示形式（例如，基于点的表示形式）不同，顶点的笛卡尔坐标被忽略，局部和非局部特征是位置不敏感的，从而可以更好地概括形状特征，并促进不变性至相似性变换。我们强调，我们仅将顶点位置用于显示正在演化的网络，但是它们的位置对任务没有影响。

### 6.2 空间自适应池化

我们基于学习到的边特征，开发了通过边收缩进行的池化操作，从而引出了由网络损失函数指导的任务驱动池化。可视化网络确定的一系列重要功能，有助于获得关于网络实质学习内容的见解。研究这种强大的机制可能会导致对神经网络行为的更好理解。网格池化为模型的可解释性和泛化性提供了有用的见解，我们相信可以为现实中的用例提供更好的线索。可以使用严格的几何边收缩（例如，二次边收缩）来代替网络学习的边收缩，它为每个网格提供了预先固定的简化形式（无论损失函数或网络权重如何）。但是，如果简化不是由网络动态确定的，则它将无法提供与我们在图 1, 8, 9 和 11 中显示的相同的独特视觉洞察。



### 6.3 局限性和未来工作

未来工作的机会可能是将空间自适应的不规则任务驱动的池化应用于基于图像的 CNN 任务。例如，高分辨率图像分割通常会产生低分辨率分割图并对其进行上采样，可能使用跳跃连接。MeshCNN 中的池化在语义上简化了具有统一特征的区域，同时保留了复杂的特征；因此，在将来，我们有兴趣对图像分割任务应用类似的不规则池化以获取高分辨率分割图，其中图像中的大均匀区域将由少量三角形表示。

当前，我们的实现是执行串行的边收缩。通过对边特征使用并行排序技术可能在 GPU 上并行执行此操作（每个池化操作仅计算一次），并确保仅同时收缩不相邻的边。显然，以非串行方式池化特征可能与串行方式有所不同。

尽管我们的方法对不同的三角剖分具有鲁棒性（如我们的实验所示），但 MeshCNN 像其他任何网络一样，都依赖良好的训练数据来成功进行泛化。从这个意义上讲，就像图像中的对抗性噪声一样，MeshCNN 容易受到对抗性重划分攻击，这可能会阻碍性能。因此，对抗攻击的鲁棒性是未来工作的有趣方向。

未来研究的另一种途径是生成模型，网格上采样和属性合成，以修改现有网络。我们的想法是通过记下边收缩列表，以与边收缩操作相反的顺序应用顶点分割。因此，当合成新的网格时，网络决定分割哪个顶点，例如，通过分割与具有高特征值的边相邻的顶点。

最后，我们将为三角形网格设计提出的策略扩展到一般图形，这是一个有前途的工作。基于边收缩的池化和上池化可以以与我们提出的 MeshCNN 类似的方式应用于一般图。至于卷积，我们必须考虑适合于一般图的不规则形的适当替代方案。一种有趣的方法可能是使用注意力机制处理边。

## 参考文献

- [1] Hanocka, Rana, et al. "MeshCNN: a network with an edge." *ACM Transactions on Graphics (TOG)* 38.4 (2019): 90.
- [2] Adobe. 2016. Adobe Fuse 3D Characters. <https://www.mixamo.com>.
- [3] Dragomir Anguelov , Praveen Srinivasan , Daphne Koller , Sebastian Thrun , Jim Rodgers , James Davis, SCAPE: shape completion and animation of people, *ACM SIGGRAPH 2005 Papers*, July 31-August 04, 2005, Los Angeles, California [doi>10.1145/1186822.1073207]
- [4] James Atwood , Don Towsley, Diffusion-convolutional neural networks, *Proceedings of the 30th International Conference on Neural Information Processing Systems*, p.2001-2009, December 05-10, 2016, Barcelona, Spain
- [5] Matan Atzmon , Haggai Maron , Yaron Lipman, Point convolutional neural networks by extension operators, *ACM Transactions on Graphics (TOG)*, v.37 n.4, August 2018 [doi>10.1145/3197517.3201301]
- [6] Mark de Berg , Otfried Cheong , Marc van Kreveld , Mark Overmars, *Computational Geometry: Algorithms and Applications*, Springer-Verlag TELOS, Santa Clara, CA, 2008
- [7] Federica Bogo , Javier Romero , Matthew Loper , Michael J. Black, FAUST: Dataset and Evaluation for 3D Mesh Registration, *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, p.3794-3801, June 23-28, 2014 [doi>10.1109/CVPR.2014.491]
- [8] Davide Boscaini, Jonathan Masci, Simone Melzi, Michael M Bronstein, Umberto Castellani, and Pierre Vandergheynst. 2015. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. In *Computer Graphics Forum*, Vol. 34. Wiley Online Library, 13--23.
- [9] Davide Boscaini , Jonathan Masci , Emanuele Rodoià , Michael Bronstein, Learning shape correspondence with anisotropic convolutional neural networks, *Proceedings of the 30th International Conference on Neural Information Processing Systems*, p.3197-3205, December 05-10, 2016, Barcelona, Spain
- [10] Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, and Bruno Lévy. 2010. *Polygon mesh processing*. AK Peters/CRC Press.
- [11] Darko Bozidar and Tomaz Dobravec. 2015. Comparison of parallel sorting algorithms. *CoRR* abs/1511.03404 (2015).
- [12] Andrew Brock, Theodore Lim, J.M. Ritchie, and Nick Weston. 2016. Generative and Discriminative Voxel Modeling with Convolutional Neural Networks. In *NIPS 3D Deep Learning Workshop*.
- [13] Alexander M. Bronstein , Michael M. Bronstein , Leonidas J. Guibas , Maks Ovsjanikov, Shape google: Geometric words and expressions for invariant shape retrieval, *ACM Transactions on Graphics (TOG)*, v.30 n.1, p.1-20, January 2011 [doi>10.1145/1899404.1899405]
- [14] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. 2017. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Process. Mag.* 34, 4 (2017), 18--42.
- [15] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2014. Spectral Networks

- and Locally Connected Networks on Graphs. In International Conference on Learning Representations (ICLR).
- [16] C. Cangea, P. Velickovic, N. Jovanovic, T. Kipf, and P. Lio. 2018. Towards Sparse Hierarchical Graph Classifiers. In NeurIPS Workshop on Relational Representation Learning.
  - [17] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2018. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence* 40, 4 (2018), 834--848.
  - [18] Michaël Defferrard, Xavier Bresson, Pierre Vandergheynst, Convolutional neural networks on graphs with fast localized spectral filtering, *Proceedings of the 30th International Conference on Neural Information Processing Systems*, p.3844-3852, December 05-10, 2016, Barcelona, Spain
  - [19] Danielle Ezuz, Justin Solomon, Vladimir G. Kim, Mirela Ben-Chen, GWCNN: A Metric Alignment Layer for Deep Shape Analysis, *Computer Graphics Forum*, v.36 n.5, p.49-57, August 2017 [doi>10.1111/cgf.13244]
  - [20] Xifeng Gao, Daniele Panozzo, Wenping Wang, Zhigang Deng, Guoning Chen, Robust structure simplification for hex re-meshing, *ACM Transactions on Graphics (TOG)*, v.36 n.6, November 2017 [doi>10.1145/3130800.3130848]
  - [21] Michael Garland, Paul S. Heckbert, Surface simplification using quadric error metrics, *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, p.209-216, August 1997 [doi>10.1145/258734.258849]
  - [22] Daniela Giorgi, Silvia Biasotti, and Laura Paraboschi. 2007. Shape retrieval contest 2007: Watertight models track. *SHREC competition 8*, 7 (2007).
  - [23] Francisco Gomez-Donoso, Alberto Garcia-Garcia, J Garcia-Rodriguez, Sergio Orts-Escolano, and Miguel Cazorla. 2017. Lonchanet: A sliced-based cnn architecture for real-time 3d object recognition. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 412--418.
  - [24] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 2017. 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. *CoRR abs/1711.10275* (2017).
  - [25] Paul Guerrero, Yanir Kleiman, Maks Ovsjanikov, and Niloy J. Mitra. 2018. PCPNet: Learning Local Shape Properties from Raw Point Clouds. *Computer Graphics Forum* 37, 2 (2018), 75–85. <https://doi.org/10.1111/cgf.13343>
  - [26] Niv Haim, Nimrod Segol, Heli Ben-Hamu, Haggai Maron, and Yaron Lipman. 2018. Surface Networks via General Covers. *CoRR abs/1812.10705* (2018).
  - [27] Rana Hanocka, Noa Fish, Zhenhua Wang, Raja Giryes, Shachar Fleishman, Daniel Cohen-Or, ALIGNet: Partial-Shape Agnostic Alignment via Unsupervised Learning, *ACM Transactions on Graphics (TOG)*, v.38 n.1, p.1-14, February 2019 [doi>10.1145/3267347]
  - [28] Mikael Henaff, Joan Bruna, and Yann LeCun. 2015. Deep Convolutional Networks on Graph-Structured Data. *CoRR abs/1506.05163* (2015).
  - [29] Hugues Hoppe, View-dependent refinement of progressive meshes, *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, p.189-198, August 1997 [doi>10.1145/258734.258843]
  - [30] Hugues Hoppe, New quadric metric for simplifying meshes with appearance attributes,

- Proceedings of the conference on Visualization '99: celebrating ten years, p.59-66, October 1999, San Francisco, California, USA
- [31] Hugues Hoppe , Tony DeRose , Tom Duchamp , John McDonald , Werner Stuetzle, Mesh optimization, Proceedings of the 20th annual conference on Computer graphics and interactive techniques, p.19-26, August 02-06, 1993, Anaheim, CA  
[doi>10.1145/166117.166119]
  - [32] Yangqing Jia. 2014. Learning Semantic Image Representations at a Large Scale. (2014).
  - [33] Chiyu Max Jiang, Jingwei Huang, Karthik Kashinath, Prabhat, Philip Marcus, and Matthias Niessner. 2019. Spherical CNNs on Unstructured Grids. In International Conference on Learning Representations. <https://openreview.net/forum?id=Bkl-43C9FQ>
  - [34] Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, and Siddhartha Chaudhuri. 2017. 3D shape segmentation with projective convolutional networks. In Proc. CVPR, Vol. 1. 8.
  - [35] Evangelos Kalogerakis , Aaron Hertzmann , Karan Singh, Learning 3D mesh segmentation and labeling, ACM Transactions on Graphics (TOG), v.29 n.4, July 2010  
[doi>10.1145/1778765.1778839]
  - [36] Michael M. Bronstein, Intrinsic shape context descriptors for deformable shapes, Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p.159-166, June 16-21, 2012
  - [37] Ilya Kostrikov, Zhongshi Jiang, Daniele Panozzo, Denis Zorin, and Burna Joan. 2018. Surface Networks. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
  - [38] Alex Krizhevsky , Ilya Sutskever , Geoffrey E. Hinton, ImageNet classification with deep convolutional neural networks, Proceedings of the 25th International Conference on Neural Information Processing Systems, p.1097-1105, December 03-06, 2012, Lake Tahoe, Nevada
  - [39] Longin Jan Latecki , Rolf Lakämper, Shape Similarity Measure Based on Correspondence of Visual Parts, IEEE Transactions on Pattern Analysis and Machine Intelligence, v.22 n.10, p.1185-1190, October 2000 [doi>10.1109/34.879802]
  - [40] Longin Jan Latecki , Rolf Lakämper, Shape Similarity Measure Based on Correspondence of Visual Parts, IEEE Transactions on Pattern Analysis and Machine Intelligence, v.22 n.10, p.1185-1190, October 2000 [doi>10.1109/34.879802]
  - [41] Yann LeCun, Learning invariant feature hierarchies, Proceedings of the 12th international conference on Computer Vision, October 07-13, 2012, Florence, Italy [doi>10.1007/978-3-642-33863-2\_51]
  - [42] Yangyan Li, Rui Bu, Mingchao Sun, and Baoquan Chen. 2018. PointCNN. CoRR abs/1801.07791 (2018).
  - [43] Yangyan Li , Sören Pirk , Hao Su , Charles R. Qi , Leonidas J. Guibas, FPN: field probing neural networks for 3D data, Proceedings of the 30th International Conference on Neural Information Processing Systems, p.307-315, December 05-10, 2016, Barcelona, Spain
  - [44] Z. Lian , A. Godil , B. Bustos , M. Daoudi , J. Hermans , S. Kawamura , Y. Kurita , G. Lavoué , H. V. Nguyen , R. Ohbuchi , Y. Ohkita , Y. Ohishi , F. Porikli , M. Reuter , I. Sipiran , D. Smeets , P. Suetens , H. Tabia , D. Vandermeulen, SHREC'11 track: shape retrieval on non-rigid 3D watertight meshes, Proceedings of the 4th Eurographics conference on 3D Object Retrieval, April 10, 2011, Llandudno, UK
  - [45] Or Litany, Alexander M. Bronstein, Michael M. Bronstein, and Ameesh Makadia. 2018.

- Deformable Shape Completion With Graph Convolutional Autoencoders. In CVPR.
- [46] Haggai Maron , Meirav Galun , Noam Aigerman , Miri Trope , Nadav Dym , Ersin Yumer , Vladimir G. Kim , Yaron Lipman, Convolutional neural networks on surfaces via seamless toric covers, *ACM Transactions on Graphics (TOG)*, v.36 n.4, July 2017  
[doi>10.1145/3072959.3073616]
  - [47] Jonathan Masci , Davide Boscaini , Michael M. Bronstein , Pierre Vandergheynst, Geodesic Convolutional Neural Networks on Riemannian Manifolds, *Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, p.832-840, December 07-13, 2015 [doi>10.1109/ICCVW.2015.112]
  - [48] Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. 2017. Geometric deep learning on graphs and manifolds using mixture model CNNs. In *Proc. CVPR*, Vol. 1. 3.
  - [49] Federico Monti, Oleksandr Shchur, Aleksandar Bojchevski, Or Litany, Stephan Gunnemann, and Michael M. Bronstein. 2018. Dual-Primal Graph Convolutional Networks. *CoRR* abs/1806.00770 (2018).
  - [50] Mathias Niepert , Mohamed Ahmed , Konstantin Kutzkov, Learning convolutional neural networks for graphs, *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, June 19-24, 2016, New York, NY, USA
  - [51] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *NIPS-W*.
  - [52] Adrien Poulénard , Maks Ovsjanikov, Multi-directional geodesic neural networks via equivariant convolution, *ACM Transactions on Graphics (TOG)*, v.37 n.6, November 2018  
[doi>10.1145/3272127.3275102]
  - [53] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR)*, IEEE 1, 2 (2017), 4.
  - [54] Charles R. Qi, Hao Su, Matthias Niessner, Angela Dai, Mengyuan Yan, and Leonidas J. Guibas. 2016. Volumetric and multi-view CNNs for object classification on 3d data. In *Computer Vision and Pattern Recognition (CVPR)*. 5648--5656.
  - [55] Charles R. Qi , Li Yi , Hao Su , Leonidas J. Guibas, PointNet++: deep hierarchical feature learning on point sets in a metric space, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, p.5105-5114, December 04-09, 2017, Long Beach, California, USA
  - [56] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. 2018. Generating 3D faces using Convolutional Mesh Autoencoders. In *European Conference on Computer Vision (ECCV)*. Springer International Publishing, 725--741.
  - [57] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. 2017. OctNet: Learning deep 3D representations at high resolutions. In *Computer Vision and Pattern Recognition (CVPR)*.
  - [58] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234--241.
  - [59] Szymon Rusinkiewicz , Marc Levoy, QSplat: a multiresolution point rendering system for large meshes, *Proceedings of the 27th annual conference on Computer graphics and*

- interactive techniques, p.343-352, July 2000 [doi>10.1145/344779.344940]
- [60] Kripasindhu Sarkar, Basavaraj Hampiholi, Kiran Varanasi, and Didier Stricker. 2018. Learning 3D Shapes as Multi-Layered Height-maps using 2D Convolutional Networks. In Proceedings of the European Conference on Computer Vision (ECCV). 71--86.
- [61] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. 2013. Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013).
- [62] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014).
- [63] Ayan Sinha, Jing Bai, and Karthik Ramani. 2016. Deep learning 3D shape surfaces using geometry images. In European Conference on Computer Vision. Springer, 223--240.
- [64] Hang Su , Subhransu Maji , Evangelos Kalogerakis , Erik Learned-Miller, Multi-view Convolutional Neural Networks for 3D Shape Recognition, Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), p.945-953, December 07-13, 2015 [doi>10.1109/ICCV.2015.114]
- [65] F. P. Such, S. Sah, M. A. Dominguez, S. Pillai, C. Zhang, A. Michael, N. D. Cahill, and R. Ptucha. 2017. Robust Spatial Filtering With Graph Convolutional Neural Networks. IEEE Journal of Selected Topics in Signal Processing 11, 6 (Sept 2017), 884--896.
- [66] Marco Tarini, Nico Pietroni, Paolo Cignoni, Daniele Panozzo, and Enrico Puppo. 2010. Practical quad mesh simplification. In Computer Graphics Forum, Vol. 29. Wiley Online Library, 407--418.
- [67] Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, and Qian-Yi Zhou. 2018. Tangent Convolutions for Dense Prediction in 3D. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3887--3896.
- [68] Lyne P. Tchapmi, Christopher B. Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. 2017. SEGCloud: Semantic Segmentation of 3D Point Clouds. In 3DV.
- [69] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph Attention Networks. In International Conference on Learning Representations.
- [70] Nitika Verma, E. Boyer, and Jakob Verbeek. 2018. FeaStNet: Feature-Steered Graph Convolutions for 3D Shape Analysis. In CVPR.
- [71] Daniel Vlasic , Ilya Baran , Wojciech Matusik , Jovan Popović, Articulated mesh animation from multi-view silhouettes, ACM Transactions on Graphics (TOG), v.27 n.3, August 2008 [doi>10.1145/1360612.1360696]
- [72] Peng-Shuai Wang , Yang Liu , Yu-Xiao Guo , Chun-Yu Sun , Xin Tong, O-CNN: octree-based convolutional neural networks for 3D shape analysis, ACM Transactions on Graphics (TOG), v.36 n.4, July 2017 [doi>10.1145/3072959.3073608]
- [73] Yunhai Wang , Shmulik Asafi , Oliver van Kaick , Hao Zhang , Daniel Cohen-Or , Baoquan Chen, Active co-analysis of a set of shapes, ACM Transactions on Graphics (TOG), v.31 n.6, November 2012 [doi>10.1145/2366145.2366184]
- [74] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. 2018a. Dynamic graph CNN for learning on point clouds. arXiv preprint arXiv:1801.07829 (2018).
- [75] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M

- Solomon. 2018b. Dynamic Graph CNN for Learning on Point Clouds. arXiv preprint arXiv:1801.07829 (2018).
- [76] Francis Williams, Teseo Schneider, Claudio Silva, Denis Zorin, Joan Bruna, and Daniele Panozzo. 2018. Deep Geometric Prior for Surface Reconstruction. arXiv preprint arXiv:1811.10943 (2018).
- [77] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3D shapenets: A deep representation for volumetric shapes. In Computer Vision and Pattern Recognition (CVPR). 1912--1920.
- [78] Haotian Xu, Ming Dong, and Zichun Zhong. 2017. Directionally Convolutional Networks for 3D Shape Segmentation. In Proceedings of the IEEE International Conference on Computer Vision. 2698--2707.
- [79] Li Yi, Hao Su, Xingwen Guo, and Leonidas Guibas. 2017. SyncSpecCNN: Synchronized Spectral CNN for 3D Shape Segmentation. In Computer Vision and Pattern Recognition (CVPR).
- [80] Rex Ying , Jiaxuan You , Christopher Morris , Xiang Ren , William L. Hamilton , Jure Leskovec, Hierarchical graph representation learning with differentiable pooling, Proceedings of the 32nd International Conference on Neural Information Processing Systems, p.4805-4815, December 03-08, 2018, Montréal, Canada
- [81] Tinghui Zhou , Richard Tucker , John Flynn , Graham Fyffe , Noah Snavely, Stereo magnification: learning view synthesis using multiplane images, ACM Transactions on Graphics (TOG), v.37 n.4, August 2018 [doi>10.1145/3197517.3201323]

## 附录

### 神经网络训练配置

对于分类，我们对 SHREC 和 Cube 雕刻数据集（从 750 条边开始）使用相同的网络结构。对于 COSEG 和人体数据集的分割任务，我们使用 Ronneberger 等人提出的 Unet 型网络（从 2250 条边开始）。我们在表 6 中详细介绍了网络配置和学习参数。

表 5 网络配置。分割网络具有一个对称的上行网络。使用 Adam 优化，学习率为 0.0002，组归一化参数为 16。数据增强使用的是 5% 的边翻转和 20% 的顶点扰动。

Classification	Segmentation (Down)
MeshConv $f_{in} \times 32$	ResConv $f_{in} \times 32$
MeshPool $\rightarrow 600$	MeshPool $\rightarrow 1800$
MeshConv $32 \times 64$	ResConv $32 \times 64$
MeshPool $\rightarrow 450$	MeshPool $\rightarrow 1350$
MeshConv $64 \times 128$	ResConv $64 \times 128$
MeshPool $\rightarrow 300$	MeshPool $\rightarrow 600$
MeshConv $128 \times 256$	ResConv $128 \times 256$
MeshPool $\rightarrow 279$	
GlobalAvgPool	
FC $256 \times 100$	
FC $100 \times 30$	



