

# 中国科学技术大学

# 博士学位论文



## 适用于深度几何学习的 几何形状表示方式研究

作者姓名： 冯万泉

学科专业： 计算数学

导师姓名： 张举勇 教授

完成时间： 二〇二二年三月二十三日



University of Science and Technology of China  
A dissertation for doctor's degree



**Research on Geometric Shape  
Representation for Deep Geometric  
Learning**

Author: Wanquan Feng

Speciality: Computational Mathematics

Supervisor: Prof. Juyong Zhang

Finished time: March 23, 2022



## 中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文，是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外，论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名：\_\_\_\_\_

签字日期：\_\_\_\_\_

## 中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一，学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权，即：学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

控阅的学位论文在解密后也遵守此规定。

公开  控阅（\_\_\_\_ 年）

作者签名：\_\_\_\_\_

导师签名：\_\_\_\_\_

签字日期：\_\_\_\_\_

签字日期：\_\_\_\_\_



## 摘要

在三维几何研究领域，几何表示是重要的研究课题。对三维几何对象来说，不同的三维表示可能适合不同的任务情形；反过来，一些特定的任务也需要提出有针对性的三维几何表示。首先，对于单个的物体，如何表示其三维形状有多种方式，如网格、点云、体素、多视角图像、深度图像等等。其次，对于有潜在对应关系的多个三维物体，如何表示其间的形变使其合理配准是一个重要的问题，这对于很多衍生任务都很关键，如跟踪、重建等。针对上述的单一物体和多物体间配准两种情形，本文提出了一系列适合深度神经网络的三维几何表示，主要包含以下几个方面：

- 基于神经场的点云局部隐式表示；
- 基于径向基函数的局部形状表示；
- 基于逐点加权刚性形变的非刚性几何形变表示；
- 基于可微形变节点图的非刚性形变表示。

除此以外，本文也在相应的任务上进行实验，探讨了所提出的上述几种几何表示的优越性。

**基于神经场的点云局部隐式表示：**传统点云表示只将每个点表示为三维空间中的一个位置或局部平面。本文提出了一种基于神经场的点云局部隐式表示，使得其中每个点通过神经场表示一个局部连续的几何形状。该表示是用包含丰富几何细节的曲面训练的，这样训练出来的模型对各种形状都有足够的表达能力。具体来说，本文提取点上的深度局部特征，并通过二维参数域和三维局部小块之间的局部同构来构建神经场。最后，局部神经场被整合在一起形成全局曲面。实验结果表明，神经点云具有强大的表达能力、鲁棒性和泛化性。据此，本文可以以任意分辨率对点云进行重采样，并且优于之前的点云上采样方法。

**基于径向基函数的局部形状表示：**将传统三维几何表示用于深度学习需要面临一些挑战，其中非规整性和高复杂度是重要的困难，这在三维模型本身具有丰富几何细节的情况下尤其明显。本文关注三维模型的细节恢复问题，使得在低分辨率模型中不能很好地表示的几何细节可以在生成的高质量模型中恢复并很好地表示。为此，本文提出了基于径向基函数的局部形状表示，用隐式表示和分治策略分别克服非规整性和高复杂度。为了训练，本文构建了一个由真实和合成扫描模型组成的数据集。实验表明，本文提出的表示具有良好的效果和泛化性。

**基于逐点加权刚性形变的非刚性几何形变表示：**非刚性配准问题由于其高自由度，在之前很难以端到端的方式在神经网络中进行学习。为此，本文提出了一种新的形变表示，用刚性变换的逐点组合来拟合非刚性变换。这不仅使解空间受

到很好的约束，而且能够使用循环框架迭代求解，从而大大降低了学习的难度。除此以外，没有足够的带标签数据是非刚性配准的另一大难点，为此本文引入了一个可微的无监督损失函数，将三维曲面投影到多视角二维图像上来衡量三维形状相似性，进行无监督端到端的训练，并取得优于以前的方法的效果。

**基于可微形变节点图的非刚性形变表示：**传统的非刚性配准方法往往通过优化的方式迭代地更新对应关系和非刚性形变。在算法流程中，对应关系的构造和迭代方式是结果的关键，而传统优化策略可能会导致局部最优。本文参照传统优化方法中的模式，提出一种可以在神经网络中使用的基于可微形变节点图的非刚性形变表示，使其端到端可训练。具体来说，本文设计了一个神经网络来预测对应关系及置信度，用可微节点图优化形变，再采用基于门循环单元的循环网络进行迭代细化，最终再预测顶点级别的偏移量，做到精细配准。该模型可以自监督训练，并取得了优于之前方法的效果。

综上，本文提出四种不同的几何表示方式，他们都是适用于深度几何学习的，而且在相应的任务上都取得了相较于之前的方法更加优越的效果。具体的相应任务包括：三维点云的上采样、三维模型的几何细节恢复、三维曲面的非刚性配准。

**关键词：**三维几何表示；局部隐式表示；径向基函数；神经场；循环网络；可微渲染；可微节点图

## ABSTRACT

In the 3D geometry research area, the geometric representation is an important research topic. For 3D geometric objects, the 3D representations suitable for different tasks may be different; conversely, each task also needs its appropriate geometric representation. First, for a single object, there are sever ways to represent its 3D shape, such as meshes, point clouds, voxels, multi-view images, depth images, etc. Second, for multiple 3D objects with potential correspondences, representing the deformations between them for reasonable registration is an important issue, which is critical for many downstream tasks, such as tracking, reconstruction, etc. For the above two cases of a single object and multi-object registration, this paper proposes a series of 3D geometric representations suitable for deep neural networks, mainly including the following aspects:

- Neural field based local implicit representation of point cloud;
- Radial basis function based local shape representation;
- Non-rigid deformation representation based on point-wise weighted combination of rigid transformations;
- Differentiable deformation graph based non-rigid deformation representation.

In addition, this paper also conducts experiments on corresponding tasks to explore the superiority of the proposed geometric representations above.

**Neural field based local implicit representation of point cloud:** Traditional point cloud only represents each point as a position or a local plane in the 3D space. In this paper, we propose a local implicit point cloud representation, namely Neural Points, where each point represents a local continuous geometric shape via neural fields. Neural Points is trained with surfaces containing rich geometric details, such that the trained model has enough expression ability for various shapes. Specifically, we extract deep local features on the points and construct neural fields through the local isomorphism between the 2D parametric domain and the 3D local patch. In the final, local neural fields are integrated together to form the global surface. Experimental results show that Neural Points has powerful representation ability and demonstrate excellent robustness and generalization ability. In this manner, we can resample point cloud with arbitrary resolutions, and it outperforms state-of-the-art point cloud upsampling methods by a large margin.

**Radial basis function based local shape representation:** The traditional 3D ge-

---

## Abstract

---

ometric representations in the deep learning pipeline face some challenges for complex 3D models, in which non-regularity and high complexity are essential difficulties. This paper focuses on the detail recovery task of 3D models so that geometric details that are not well represented in low-resolution models can be recovered and well represented in the result high-quality models. To this end, this paper proposes a local shape representation based on radial basis functions and designs a deep network to solve this problem and overcome the above challenges. We apply the implicit representation and the divide-and-conquer strategies to overcome the non-regularity and high complexity, respectively. To train the network, we constructed a dataset consisting of real and synthetic scan models, including high/low-quality pairs. Experimental results show that our proposed representation and network are suitable for general models and outperform previous methods in recovering geometric details.

**Non-rigid deformation representation based on point-wise weighted combination of rigid transformations:** Learning non-rigid registration in an end-to-end manner is challenging due to the inherent high degrees of freedom and the lack of labeled training data. In this paper, we resolve these two challenges simultaneously. First, we propose to represent the non-rigid transformation with a point-wise combination of several rigid transformations. This representation not only makes the solution space well-constrained but also enables our method to be solved iteratively with a recurrent framework, which greatly reduces the difficulty of learning. Second, we introduce a differentiable loss function that measures the 3D shape similarity on the projected multi-view 2D depth images so that our full framework can be trained end-to-end without ground truth supervision. Extensive experiments on several different datasets demonstrate that our proposed method outperforms the previous state-of-the-art by a large margin.

**Differentiable deformation graph based non-rigid deformation representation:** The traditional pipeline for non-rigid registration is to iteratively update the correspondence and alignment such that the transformed source surface aligns well with the target surface. Among the pipeline, the correspondence construction and iterative manner are key to the results, while existing strategies might result in local optima. In this paper, we adopt the widely used deformation graph based representation, while replacing some key modules with neural learning based strategies. Specifically, we design a neural network to predict the correspondence and its reliability confidence rather than the strategies like nearest neighbor search and pair rejection. Besides, we adopt the GRU-based recurrent network for iterative refinement, which is more robust than the

---

## Abstract

---

traditional strategy. The model is trained in a self-supervised manner, and thus can be used for arbitrary datasets without ground-truth. Extensive experiments demonstrate that our proposed method outperforms the state-of-the-art methods by a large margin.

To sum up, we propose four different geometric representations, all of which are suitable for deep geometric learning, and have achieved better results than previous methods on the related tasks. Specific related tasks include: point cloud upsampling, geometric detail recovery of 3D models, and non-rigid registration of 3D surfaces.

**Key Words:** 3D Geometry Representation; Local Implicit Function; Radial Basis Function; Neural Field; Recurrent Network; Differentiable Rendering; Differentiable Deformation Graph



# 目 录

第 1 章 绪论 ······	1
1.1 问题介绍与研究价值 ······	1
1.1.1 三维视觉的重要性 ······	1
1.1.2 三维几何表示简介 ······	2
1.2 相关工作与研究现状 ······	4
1.2.1 单个物体的几何形状表示 ······	5
1.2.2 多物体间的几何形变表示 ······	14
1.3 研究内容与结构安排 ······	19
第 2 章 基于神经场的点云局部隐式表示 ······	20
2.1 引言 ······	20
2.2 基于神经场的点云局部隐式表示：设计与应用 ······	21
2.2.1 概述：基于点云的表示 ······	21
2.2.2 算法流程 ······	21
2.2.3 网络结构 ······	23
2.2.4 损失函数 ······	25
2.3 实验结果和讨论 ······	26
2.3.1 实现细节 ······	26
2.3.2 消融实验 ······	27
2.3.3 结果和比较 ······	28
2.3.4 泛化能力和鲁棒性 ······	30
2.3.5 更多结果和分析 ······	31
2.4 本章小结 ······	33
第 3 章 基于径向基函数的局部形状表示 ······	34
3.1 引言 ······	34
3.2 基于径向基函数的局部形状表示：设计与应用 ······	36
3.2.1 基于径向基函数的局部形状表示 ······	36
3.2.2 算法流程 ······	37
3.2.3 网络结构 ······	40
3.3 实验结果和讨论 ······	42
3.3.1 实现细节 ······	42
3.3.2 结果 ······	45

## 目 录

---

3.3.3 消融实验 ······	47
3.3.4 比较 ······	48
3.3.5 鲁棒性和泛化性 ······	50
3.4 本章小结 ······	53
<b>第 4 章 基于逐点加权刚性形变的非刚性几何形变表示 ······</b>	<b>54</b>
4.1 引言 ······	54
4.2 基于逐点加权刚性形变的非刚性几何形变表示：设计与应用 ······	56
4.2.1 问题描述 ······	56
4.2.2 基于逐点加权刚性形变的非刚性几何形变表示 ······	56
4.2.3 循环更新框架 ······	56
4.2.4 网络结构和损失函数 ······	57
4.3 实验结果和讨论 ······	61
4.3.1 实现细节 ······	62
4.3.2 消融实验 ······	63
4.3.3 结果和比较 ······	63
4.4 本章小结 ······	67
<b>第 5 章 基于可微形变节点图的非刚性形变表示 ······</b>	<b>70</b>
5.1 引言 ······	70
5.2 基于可微形变节点图的非刚性形变表示：设计与应用 ······	71
5.2.1 算法流程 ······	72
5.2.2 损失函数 ······	74
5.3 实验结果和讨论 ······	75
5.3.1 实现细节 ······	75
5.3.2 消融实验 ······	76
5.3.3 结果和比较 ······	77
5.4 本章小结 ······	81
<b>第 6 章 总结与展望 ······</b>	<b>83</b>
6.1 本文总结 ······	83
6.2 未来展望 ······	83
6.2.1 单个物体的几何形状表示未来展望 ······	83
6.2.2 多物体间的几何形变表示未来展望 ······	84
<b>参考文献 ······</b>	<b>85</b>
<b>致谢 ······</b>	<b>97</b>

## 目 录

在读期间发表的学术论文与取得的研究成果 · · · · · 99



# 第1章 绪论

## 1.1 问题介绍与研究价值

### 1.1.1 三维视觉的重要性

本文研究三维几何表示，这对于当下重要的研究课题，三维视觉和几何学，是至关重要的。我们生活在一个三维的世界当中，并且时刻在与这个三维的世界进行感知与交互。故而，生活中的很多关键问题，都可以归结到三维视觉领域的研究上。除此以外，幸运的是，我们当前所处的时代正处在三维视觉应用的爆发期。在当前背景下，对三维视觉与几何学的研究，注定是颇具价值的。



图 1.1 一些三维视觉的应用，依次解释如下。（a）人脸识别：超市结账刷脸支付；（b）自动驾驶：车辆自动理解场景并决定行进路线；（c）3D 游戏：近期的优秀 3D 游戏《赛博朋克 2077》；（d）体感游戏：用户佩戴传感器体验虚拟网球游戏；（e）扫描建模：使用光场扫描仪对人头进行高精度建模；（f）3D 打印：用户使用 3D 打印机 Artillery Sidewinder<sup>[1]</sup>打印手办；（g）电影动作捕捉：演员用动作捕捉的方式出演电影；（h）虚拟偶像：真人驱动的虚拟偶像“Kizuna AI”；（i）电影中的元宇宙：电影《头号玩家》中的元宇宙形态。

三维视觉的应用已经渗透到了生活的各个方面。作为一个直观的展示，本文在图1.1中对三维视觉与几何学的若干应用进行了可视化。例如，很多人现在习惯了在日常购物中使用刷脸支付的付款方式，而这主要依赖于在近几年迅速发展的三维人脸识别技术<sup>[2-3]</sup>。同时，人脸识别技术也被应用于重要的公共安全领域，例如重要公共场合的身份验证，以及通关卡口的安全检查等。再例如，无人驾驶<sup>[4-6]</sup>技术依赖于三维视觉中的物体检测和场景分析，且有着极大的可能会在将来提高民众出行的便捷性。此外，从娱乐角度来讲，随着三维视觉与图形学的发展和家用计算机算力的增强，越来越多的大型3D游戏逐渐涌现，其中使用到的渲染、模拟技术等无不与三维几何有着密切的关系。随着一批价格低廉的三维数据传感器的普及，如深度相机Kinect<sup>[7]</sup>等，我们的生活中也开始出现一些新的娱乐形式，比如体感游戏等。与此同时，我们可以用高精度的扫描设备<sup>[8]</sup>将现实生活中的物体在虚拟世界中进行高精度的重建。反过来，3D打印技术的发展也让我们有机会将虚拟的三维几何变成实际可触碰的物品，并且日趋方便、成本也日趋降低。

此外，三维视觉的一个重要的发展趋势是有希望可以作为人类未来新的沟通载体，让人摆脱传统的只有声音、2D图像或视频的形式的束缚，做到3D的通讯。此外，有一些成本不菲电影已经做出了技术落地的表率，使用了动作捕捉技术。近年来，这种技术也渐渐走入了我们的日常生活和娱乐中，并得到了新一代年轻人的认可。例如，由真人驱动的二次元虚拟主播“Kizuna AI”在国内外平台获得了火爆人气，并成功举办了虚拟演唱会。今年，基于语音-图像多模态的虚拟人主播合成视频工作也展示了另一种未来的任务方向。最后，当前业界火热的元宇宙概念也极大地依赖着三维视觉与三维几何学的发展。在业界的畅想中，用户可以身戴设备，置身虚拟的平行世界，每个人都有自己的虚拟化身，在元宇宙世界中进行自由的交互和生活，打破现实中空间的限制。

而本文的研究对象，三维几何表示，顾名思义即在三维空间中构建几何物体或场景的方式。三维几何表示在三维视觉中起着非常重要的作用，是构建三维几何世界数据的基本方式。以下将介绍三维几何表示的当前发展以及本文的研究思路。

### 1.1.2 三维几何表示简介

对于不同的具体三维视觉任务，其对表示的要求可能略有不同。通常我们需要描述一个物体的表面信息，包括表面点的位置、法向、颜色等，这时候就需要对物体的表面进行合理的建模，选择合理的三维表示来描述三维物体。除了单个物体的几何形状表示以外，有时候有些任务也要考虑多个物体间的几何形变表示，即如何将多个物体进行配准。这种需求有时会发生在要对一个可形变物

体进行形变跟踪的时候，或者是发生在想要在两个物体间建立形变的关联以达到形变驱动的效果的时候。总结来说，本文将三维几何表示分为单个物体的几何形状表示，以及多物体间的几何形变表示。下面本文将分别来分析这两种表示情形。

对于单个物体的几何形状表示，传统的表示方法包括点云、网格、体素、多视角表示、隐式函数等。例如，在上述的三维扫描建模的任务中，通常是首先用扫描仪获得点云，然后传统的流程会将其转化成体素数据，然后进行表面的提取，进而转化为网格数据。三维人脸识别的一种经典流程需要扫描人脸的深度图，然后结合图像的颜色信息一起产生由低维嵌入的参数化形式表示的人脸网格，进而执行进一步的身份判断。再例如，图像视角合成任务通常把一个场景表示为隐式场，并设计与之匹配的神经渲染方法。判断一种几何表示的优劣有多种依据，具体可以包括：对复杂几何形状的精确表达能力，良好的泛化能力，内存占用的高低，当然也包括是否适合神经网络学习。传统的表示很难兼具以上优点，而这在某些任务中就会暴露出问题。具体地，本文会提出两种新的针对单个物体的几何形状表示，并将其用在特定场景任务中，并印证他们的有效性。本文首先提出一种基于神经场的点云局部隐式表示，并将其应用在了点云连续上采样任务当中。之后本文又提出了一种基于径向基函数的局部隐式表示，并将其应用在三维模型的几何细节恢复任务当中。

对于多物体间的几何形变表示，传统的表示方法包括逐点位移场、样条函数、节点图等。尽管逐点位移场可以表示复杂的几何形变，但是其复杂度高，自由度大，这在神经网络中是非常难以学习的。另一方面，由样条函数组成的形变表示，其自由度确实是受到了很好的限制，其自由度受到空间中选取的控制点的影响。但是，这种情况下可能有些控制点离物体的表面较远，造成了不必要的浪费；而当控制点较少时却又满足不了足够表达物体目标形变的表达能力。另一种在传统方法中常用的表示是节点图，确实可以起到良好的正则化效果，在控制自由度的前提下，依然达到足够的表达能力。但是节点图的表达形式也存在一些缺点。首先就是节点图的获取比较困难；其次就是基于节点图的表示往往用于传统的优化算法中，需要使用优化的方式获取每个节点上的形变。要在神经网络中使用节点图解决非刚性形变问题并不是一件十分自然的事情。在这种情况下，本文首先提出一种基于逐点加权刚性形变的非刚性几何形变表示，并将其使用于深度循环网络中，配合了无监督的损失函数，做到了无监督端到端的训练。其次本文也提出了一种基于可微形变节点图的非刚性形变表示，在神经网络中使用了这种表示，使得其既利用了节点图的结构化信息，又是端到端可训练的。两种表示都有效提高了非刚性配准任务的效果。

总结来说，本文提出了两种针对单个物体的几何形状表示以及两种针对多

物体之间的几何形变表示，并在相应的任务上取得了相较于之前方法更好的效果，具体包括：三维点云的上采样、三维模型的几何细节恢复、三维曲面的非刚性配准。

## 1.2 相关工作与研究现状

这一章节将介绍与三维几何表示相关的若干相关工作和研究现状。具体地，本文首先在1.2.1中介绍有关单个物体的几何形状表示的相关研究工作，然后在1.2.2介绍有关多物体间形变的几何形变表示的相关研究工作。对于单个物体的几何形状表示来说，本文涉及到了点云、网格、深度图像、多视角图像这几种显式的表示，还有体素、连续隐函数等隐式表示；对于多物体间的几何形变表示来说，本文涉及到了一般刚性变换的表示，以及包括逐点偏移场、样条函数、节点图在内的非刚性形变表示。

为了做一个直观的展示，本文将其所总结到的相关三维几何表示列在图1.2中。在以下针对相关工作描述中，对每种表示，本文都会先给出其描述或定义，然后给出其在某些具体任务、工作中的应用，还有其优缺点分析。

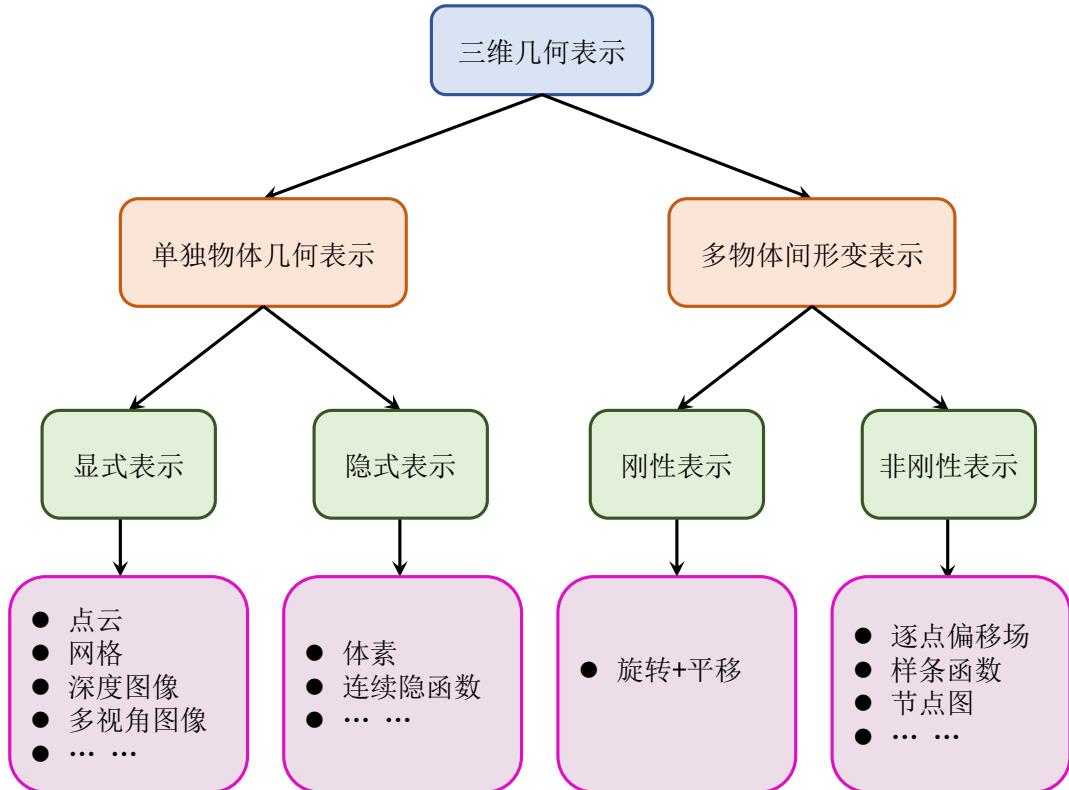


图 1.2 本章节所要介绍的相关三维表示。

### 1.2.1 单个物体的几何形状表示

三维数据是计算机视觉领域的重要研究对象，其载体三维几何表示是对数据进行访问和操作的方式。良好的几何表示应当能够表达出其所代表的数据的潜在的有关感知对象和场景的完整几何形状的丰富信息，同时也应该能够很好地应付当下的算法流程和神经网络结构。最近，随着大型三维数据集越来越容易获取，当下计算设备性能越来越高，以及不断发展的算法，我们今天通常是考虑应用深度学习来处理三维数据上的特定任务，例如分割、识别、上采样等。根据所使用的三维数据表示的不同，在某个具体的场景下、在使用不同的深度学习架构时可能会遇到不同的挑战。本章节提供了关于各种单个物体的几何形状表示的概述；根据本文的行文组织，突出了显式表示和隐式表示之间的区别。显式表示将在（1）中介绍，隐式表示将在（2）中介绍。

#### （1）显式表示

首先介绍显式表示。显式表示相对于隐式表示来说，其最大的特点是直接存储了表面的信息，而不需要额外的表面提取过程。具体地，在这一节中，本文讨论的显式表示包括：点云、网格、深度图像、多视角图像几种类型。

##### ① 点云

点云是三维视觉中另一种非常常见的三维几何表示，它将物体的表面表示为一些无序的三维点的集合。点云  $\mathcal{P}$  可以表示为：

$$\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^n, \quad (1.1)$$

其中  $\mathbf{p}_i \in \mathbb{R}^3$  是点云中具体的一个点坐标。与此同时点云也可以同时囊括颜色信息和法向信息等等逐点的信息。由于点云的形式简单，获取也很容易，所以点云的使用是较为方便的。

以点云表示为载体的几何任务，同样有很多。在用三维捕捉设备获取原始点云之后，为了提高点云数据的质量，有时候需要进行一些类似于点云降噪<sup>[9]</sup>、点云上采样<sup>[10-11]</sup>之类的工作，来方便我们进一步对点云数据进行分析、理解和操作。对于多视角的点云数据，我们往往需要寻找不同视角扫描点云之间的对应关系，以便将其进行精准的匹配来获得物体的全貌，这就涉及到了点云配准<sup>[12-13]</sup>的问题。对于能够表达有一定语义形状的点云来说，我们需要对其进行理解。当需要理解点云所表达物体种类的时候，我们就需要处理点云分类<sup>[14]</sup>任务。当想要对点云所表达物体的结构进行分析的时候，我们就需要进行点云分割<sup>[15]</sup>。在某些任务中，我们也需要将原始的点云数据转化为重建好的网格，即点云重建<sup>[16-17]</sup>。

将点云应用于神经网络当中，也是要面临挑战的。首先点云和网格类似都是非欧基里德类型的，所以我们依然没有办法使用传统的卷积神经网络来直接处理点云。同时我们也没有办法将网格上的卷积操作直接用到点云上，因为点云相

比于网格来说缺少了边的连接关系这一信息。从理论上来分析，点云是一个集合，所以说正确的神经网络操作应该要考虑到所有点的无序性和对称性。作为第一篇点云神经网络的工作，PointNet<sup>[18]</sup>提出了一种基于多层感知器（Multilayer Perceptron, MLP）和全局池化的神经网络结构，首次做到了在点云上使用深度学习。在那之后又出现了一些后续工作，包括一些在局部点云上构建边连接以实现局部卷积的网络结构，如比较经典的 DGCNN<sup>[19]</sup>等。近期随着新兴的基于注意力机制的网络 Transformer<sup>[20]</sup>的流行，也有一些近期的工作开始将 Transformer 使用到点云上并提出了基于点云的 Transformer 结构<sup>[21]</sup>。为了直观，本文在图1.3中展示了一个点云网络的示例。

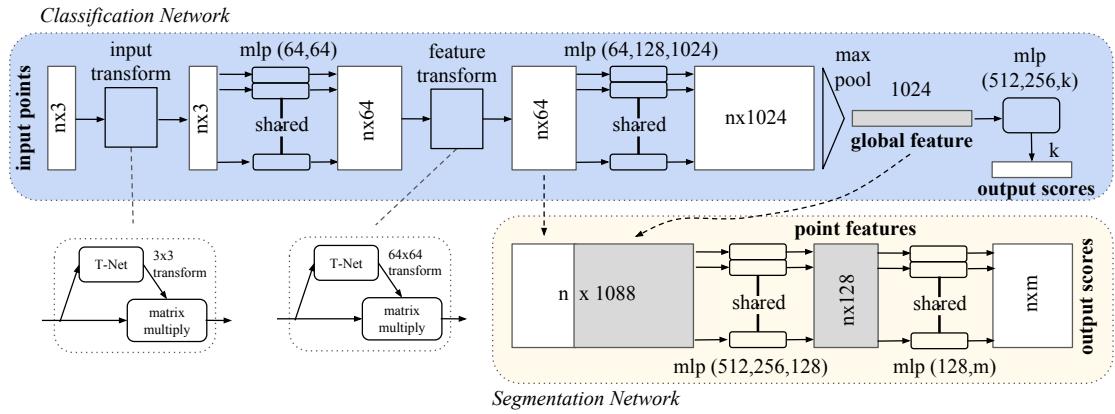


图 1.3 一个点云网络的示例。该图片引用自 PointNet 工作<sup>[18]</sup>。

点云的优点和缺点都非常的显著。点云的优点包括了点云获取的便捷性、点云形式的简单性、点云用途的多样性、点云相关研究的充分性，等等。点云的缺点也很显然，那就是没有边连接关系，缺乏面片信息。不同于网格有明确的面片连接，在点云的情况下，我们没有办法去获得物体表面的更多连续的信息，也没有办法去研究诸如表面法向、判断内外、计算符号距离值等功能。此外与网格情况类似，尽管点云可以表示物体的表面，但是其分辨率却受到了顶点数目的限制。当想要去进行一个非常高精度的表示的时候，我们就需要非常多的顶点数目，这会造成非常大的内存消耗，而且神经网络也不好处理这么大的点云。这些都会对使用点云的过程造成相应的困扰。在本文后续的章节中，本文将会提出一种神经点云表示，通过局部神经场来表示连续的点云，从而克服了分辨率的限制。

## ② 网格

网格是数字几何处理中最基本的一种数据表示方式。对于网格  $\mathcal{M}$  来说，其被表示为由顶点和边连接关系组成的图的形式，即：

$$\mathcal{M} = (\mathcal{V}, \mathcal{E}), \quad (1.2)$$

其中  $\mathcal{V}$  和  $\mathcal{E}$  分别为图的顶点集和边集。具有良好流形属性的网格可以方便地表征物体的表面，同时也可以涵盖点、面的法向和颜色信息等。

以网格为表示的三维几何任务有很多。传统的数字几何处理任务包括网格细分<sup>[22-23]</sup>，网格简化<sup>[24-25]</sup>，网格滤波<sup>[26-27]</sup>等。在这些任务中，近年均有一些基于深度学习的方法出现，例如神经网格细分（Neural Mesh Subdivision）<sup>[28]</sup>，可微曲面三角化（Differentiable Surface Triangulation）<sup>[29]</sup>等。除此之外，在三维重建任务中，如人脸重建<sup>[30]</sup>、人体重建<sup>[31-32]</sup>、场景重建<sup>[33-34]</sup>等，重建后的三维几何通常也是由网格来表示。但在以上任务中，网格通常是作为一种最终的表达形态，而非神经网络直接处理的对象。比如在人脸重建任务中，神经网络可能会首先回归其低维嵌入的参数化基（如 3DMM<sup>[35]</sup>，FaceWareHouse<sup>[36]</sup>）上的系数，然后用参数化的基加权组合出重建的网格。又或者在人体或者场景的重建中，网络可能会预测体素或者神经隐式场，然后使用行进立方体（Marching Cubes）<sup>[37]</sup>算法进行结果网格的提取。除此以外，还有一类工作是在保持网格拓扑不变的基础上对网格进行某种目标驱动下的形变，如网格的自编码器（Mesh Autoencoder）<sup>[38]</sup>相关的工作，这种工作往往是用深度学习的方法来做的。

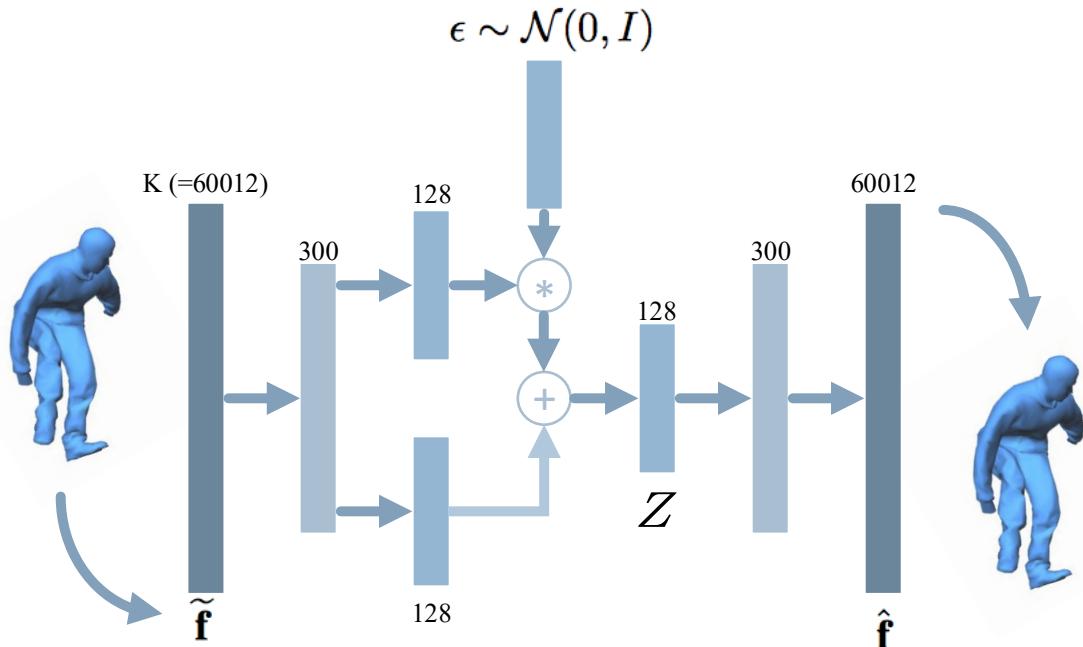


图 1.4 网格神经网络的样例，其中输入与输出的拓扑是一致的。该图片引用自网格自编码器工作<sup>[38]</sup>。

下面探讨网格这种表示与深度学习的结合。由于网格是由图结构组成的，是一种非欧几里得的表示方式，所以这也意味着我们无法直接对其实施传统的卷积神经网络。既然不容易直接把卷积神经网络用在网格上，那么相反地，许多之前的工作开始着手专门为三维网格设计相应的神经网络框架。由于网格所表示的表面可以看作三维空间中的二维嵌入，利用这一点，一些研究人员开始在三维

网格的表面尝试提取信息。这些工作将顶点之间的连接关系、面片信息纳入计算，从而定义局部网格上类似卷积的空域操作<sup>[39-41]</sup>，这样就可以有效在网格所表示的表面上提取几何特征。同时，也有一些谱域上的研究<sup>[42-43]</sup>是旨在利用图的谱特性来定义在神经网络框架中使用的内蕴描述符。这些内蕴描述符是整体的，是不受具体的非欧几里得性质的影响的，自然也是适用于深度神经网络框架当中的。

尽管图卷积网络可以有效地应用在网格上，完成特征提取，池化等网络操作，但是它也有相应的问题。图卷积网络通常是只能使用在拥有相同边连接关系和顶点数的一类网格上（本文在图1.4中展示了一个样例），但是对于顶点数和边连接关系不确定的情形没有较好的兼容能力。这就阻碍了图卷积网络的用途，尤其是在输出的网格并不能有确定顶点数和边连接关系的时候。

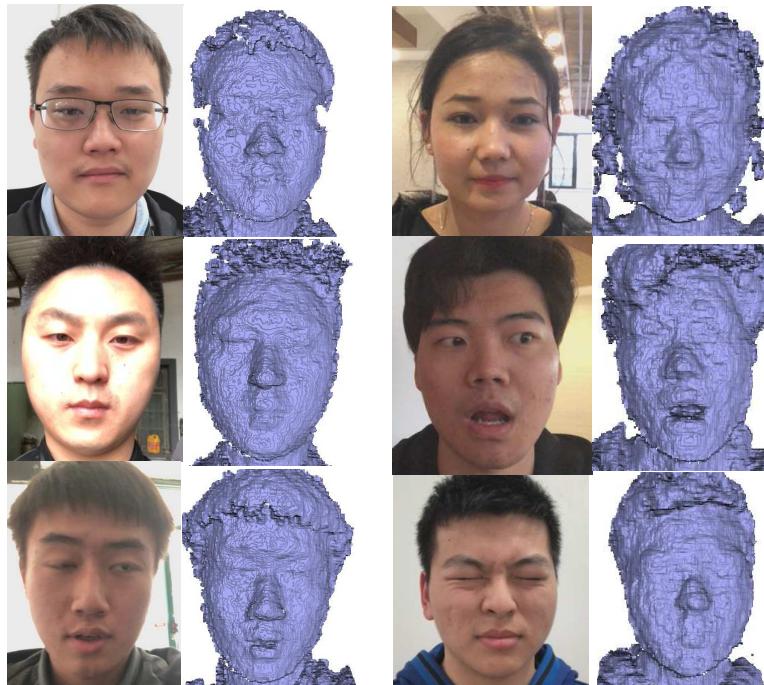


图 1.5 一些 RGBD 图像的样例。该图片引用自基于 RGBD 的人脸重建工作<sup>[30]</sup>。

### ③ 深度图像

深度图像是一种类似于灰度图像的规整二维张量，包含的每个像素值的信息为沿像素方向场景到拍摄处的实际距离，故称为深度图像，有时候也称之为“2.5D”数据。通常 RGB 图像和深度图像是同时获得的，因而是对一致场景的不同信息面的描述。随着一些深度传感器（例如微软的 Kinect<sup>[7]</sup>）的流行，我们在获取 RGB 图像的时候，也很方便顺带获取深度图像。将 RGB 图像和深度图像耦合对齐后的数据，其被称为 RGBD 图像。RGBD 数据是三维对象的简单而有效的表示，相比于普通的 RGB 图像，它包含了很多的几何信息，这对很多下游任务是很有用的。除此之外，它的获取价格也是比较低的，甚至在很多手机上已经

可以获取深度信息了，这使得我们很容易获得该种类型的数据。本文在图1.5中展示了一些 RGBD 图像的样例。

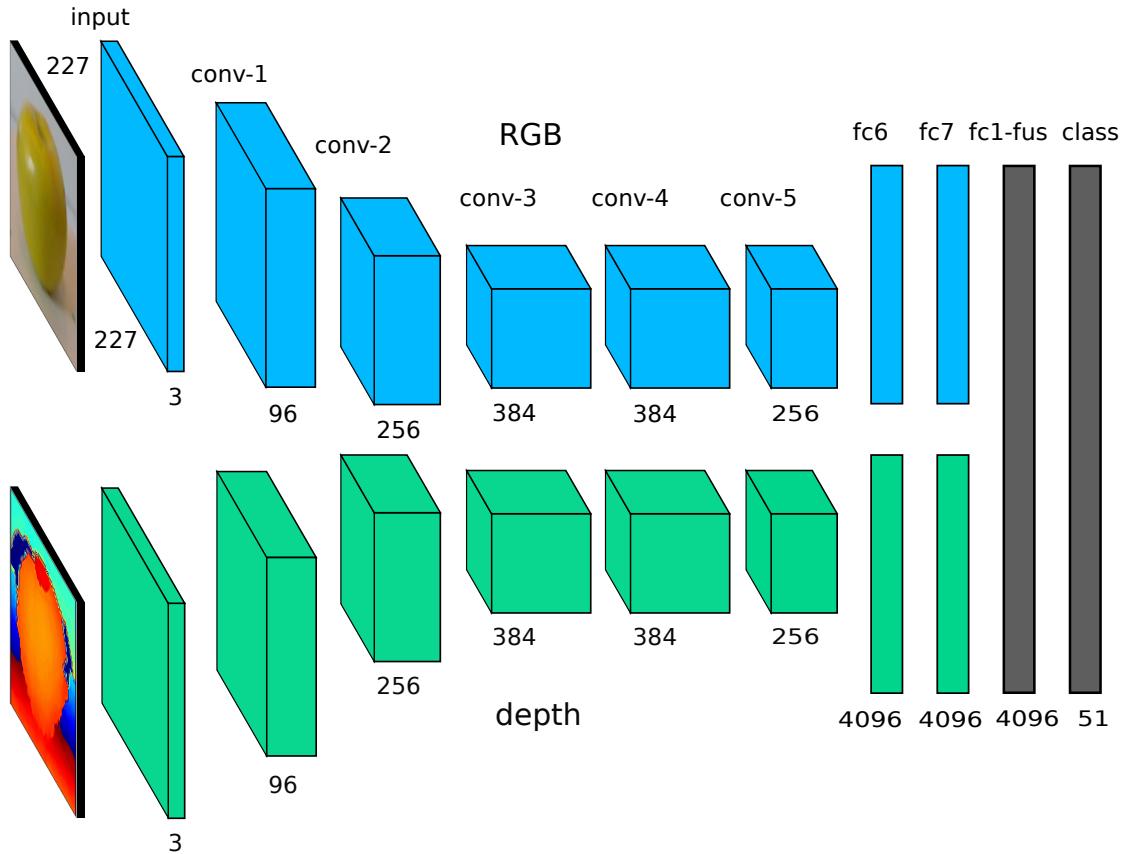


图 1.6 一个以 RGBD 为输入的神经网络结构示意图。该图片引用自 AISNet 工作<sup>[44]</sup>。

基于 RGBD 输入的算法，在很多任务上都可以获得比单纯的 RGB 更好的效果。RGBD 数据可以用于三维重建<sup>[30-31]</sup>，实时跟踪<sup>[45]</sup>等。此外也有一些工作可以通过单视角或多视角 RGB 图像来进行深度估计<sup>[5]</sup>。对于 RGBD 数据来说，在深度学习中，我们也需要具体适配的神经网络结构<sup>[44,46-47]</sup>。一般来说，一种处理 RGBD 数据的常见做法可以归纳为使用两支不同的网络分别处理深度信息和颜色信息。如图1.6所示为一个样例，展示了一个以 RGBD 为输入的神经网络结构，其中一个分支用来处理 RGB 颜色信息，另一个分支用来处理深度信息。

相比于单纯的 RGB 数据来说，多处理一个深度通道，往往是可以对算法的结果给到一些 3D 信号指导的，且这种指导往往是很有价值的。但是，深度图像并不能完整给出三维物体的几何形状，只能大致给出一些三维属性，起到辅助性的作用。毕竟深度信息并不是高质量的三维几何信息，它会受到很多现实问题的影响，包括过多的噪声、过低的分辨率以及视角的遮挡等。

#### ④ 多视角图像

基于多视角图像<sup>[48-49]</sup>的表示旨在用一个物体或场景的多个视角的图像来表示三维几何。不同于深度图需要深度相机，多视角图像表示只需要在多个视角对物体拍摄 RGB 图像即可获得，当然有的时候我们也需要对这些相机的位置进行

标注。

对多视角图像进行深度学习，要得益于二维图像卷积网络的广泛成功使用。经典的多视角图像的网络 MVCNN<sup>[49]</sup>首先对单个视角图像用卷积神经网络提取特征，然后再对各个图像的特征进行视角池化，最终将池化后的特征送入后续的网络，最终得到想要的三维特征信息。图1.7展示了 MVCNN 的网络过程。网络结构也在后续的工作<sup>[50-52]</sup>中得到了改进。

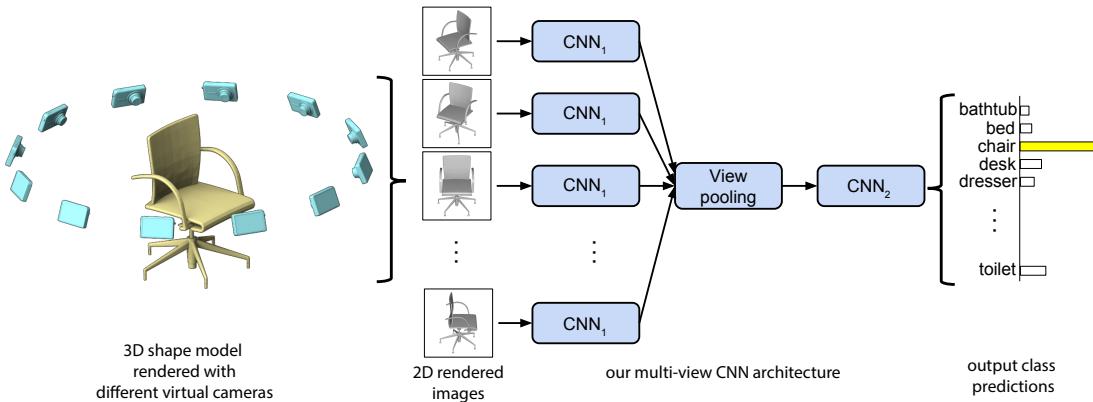


图 1.7 一个基于多视角图像表示的神经网络结构示意图。该图片引用自 MVCNN 工作<sup>[49]</sup>。

作为一种表示来说，它不如点云、网格、体素来的直接。同时，作为同样是图像类别的表示，多视角图像相对于深度图像来说，可以说一定程度上是可以克服视角遮挡的问题，也比较少受到噪声的影响。视角的数量无疑会对三维表达能力产生影响。如果视角太少，有可能不足以表达三维几何。如果视角太多又可能会导致不必要的计算开销。所以合适的视角数量是很重要的。再有就是不得不提到的一点，近两年新视角合成任务<sup>[53-54]</sup>得到了广泛的关注和飞速的发展，如图1.8所示为一个视角合成的样例示意图。尤其是以神经辐射场（Neural Radiance Fields, NeRF）<sup>[54]</sup>为核心的一系列算法，得到了快速的发展，这使得多视角图像表示渐渐在三维视觉中占据了重要的地位。

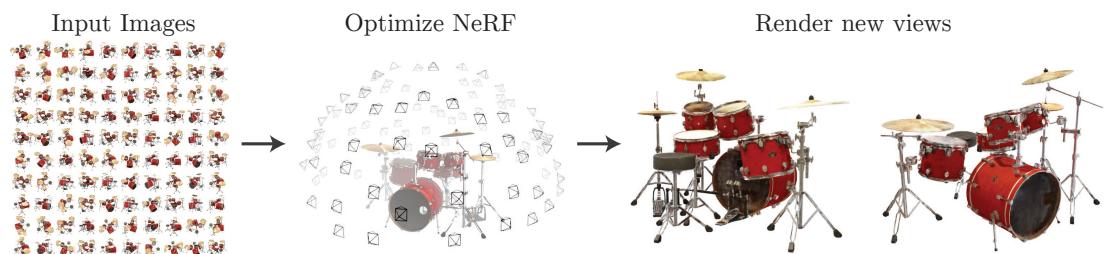


图 1.8 一个新视角合成的样例示意图。该图片引用自 NeRF 工作<sup>[54]</sup>。

## (2) 隐式表示

下面的部分将介绍隐式表示的研究现状和相关工作。相对于显式表示而言，隐式表示没有直接把物体表面的点信息表征出来。相反，隐式表示往往是建立了关于物体表面的一种隐式函数或隐式场。虽然隐式表示没有直接地记录物体表面的点坐标或者法向信息，但我们可以从中将物体表面给提取出来，从而获得例

如网格等显式的表面表示。

### ① 体素

体素是一种经典的，离散的三维隐式表示。顾名思义，体素类似于像素，是一种规整的、三维张量形式的表示。基于体素的三维几何表示，我们有时也将其简称为体表示。对于目标的物体或场景，我们假想把它置于一个三维包围盒当中。通过一定的缩放，不妨假设这个包围盒的范围被统一归一化为  $[0, 1]^3$ 。我们可以将包围盒的长宽高分别进行平均切分，从而形成很多个单元格以及格点。假如每条边被切分为到  $n$  个部分，那么单元格的个数为  $n \times n \times n$  个，格点的个数为  $(n+1) \times (n+1) \times (n+1)$  个。我们可以在这些格点上定义一些有一定几何含义的值，那么整个包围盒就成为了一个形如  $(n+1) \times (n+1) \times (n+1)$  的三维张量，即体数据。定义在格点上的标量值可以有多种选择，根据选择的不同，体表示所代表的含义也不同。在这里本文主要介绍两种经典的体素表示，分别称为符号距离场（Signed Distance Field, SDF）和占有场（Occupancy Field, OF）。

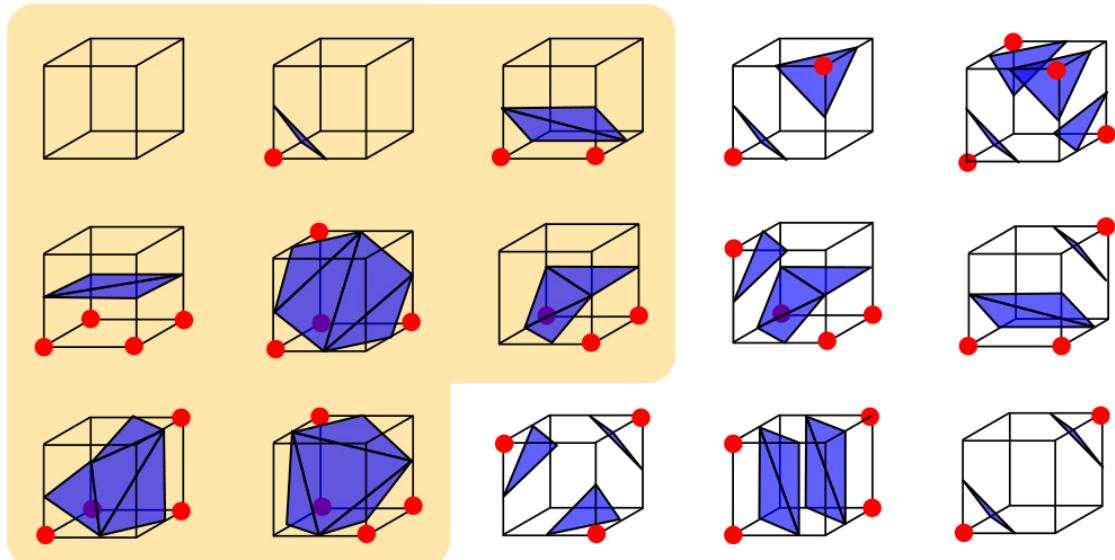


图 1.9 Marching Cubes 算法示意图。该图展示了一个单位立方体内的各种可能的拓扑情况，以及在各种拓扑情况下的面片连接方式。该图片引用自工作<sup>[17]</sup>。

设目标曲面为  $\mathcal{S}$ 。对于第  $(i, j, k)$  个单元格点位置  $(0 \leq i, j, k \leq n)$ ，不妨记该位置为  $\mathbf{x}_{i,j,k}$ 。对于符号距离场  $F_{sdf}$ ，其公式为：

$$F_{sdf}(\mathbf{x}_{i,j,k}) = \begin{cases} d(\mathbf{x}_{i,j,k}, \mathcal{S}), & \text{当 } \mathbf{x}_{i,j,k} \text{ 在 } \mathcal{S} \text{ 外部} \\ -d(\mathbf{x}_{i,j,k}, \mathcal{S}), & \text{当 } \mathbf{x}_{i,j,k} \text{ 在 } \mathcal{S} \text{ 内部} \end{cases}$$

其中  $d(\cdot, \mathcal{S})$  代表三维点到曲面  $\mathcal{S}$  的距离。可以看出，在物体表面的内外，格子点的符号距离值的符号会发生变化。对于占有场  $F_{of}$ ，其公式为：

$$F_{of}(\mathbf{x}_{i,j,k}) = \begin{cases} 0, & \text{当 } \mathbf{x}_{i,j,k} \text{ 在 } \mathcal{S} \text{ 外部} \\ 1, & \text{当 } \mathbf{x}_{i,j,k} \text{ 在 } \mathcal{S} \text{ 内部} \end{cases}$$

可以看出，可以取 1 与 0 的中间值 0.5 为一个判断值，通过相较于该判断值的大小，来推理一个格子点在  $S$  的内部或外部。对于这两种体素的表示方式，我们都可以使用 Marching Cubes 算法来进行重建，获得结果网格。Marching Cubes 重建方法如图1.9。

上述的体素表示可以表示任意拓扑，这是一般的网格表示所不具备的优点。除此以外，将体素表示用在三维神经网络也是非常容易的。因为它们的数据类型都是规整的三维张量，所以可以直接使用三维卷积来进行操作。但是，除了这些优点以外，其局限性也是非常大。首先有太多的体素格子都是远离三维物体表面的，这些体素其实是不重要甚至冗余的，但是为了规整的三维卷积操作，我们却没有办法直接单纯放弃这些部分，这就造成了内存和计算量上的浪费。再者由于三维张量相比于二维图像来说多了一维，所以当分辨率上升的时候，其复杂度的增长也是巨大的。这造成了我们用体素的方式没有办法处理超高的分辨率，从而一开始也就处于了精度上的劣势。尽管一些基于八叉树的工作<sup>[55-56]</sup>缓解了这种状况，但是其总归来说还是离散的，没有办法做到彻底突破分辨率的限制。幸运的是，分辨率的限制可以在连续隐表示中得到解决。

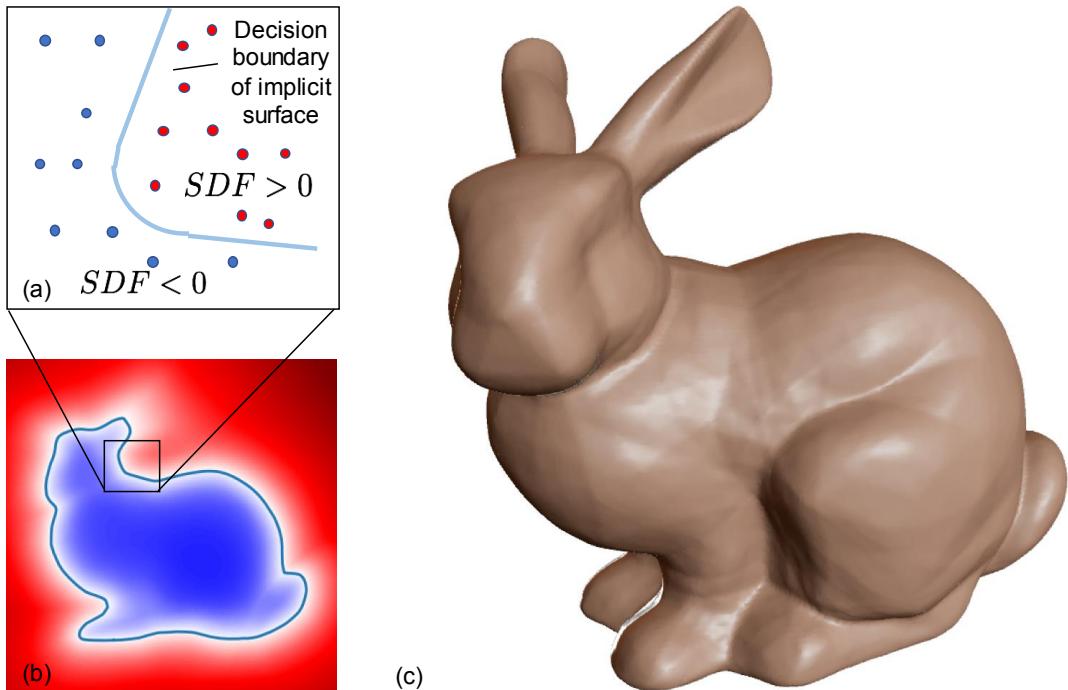


图 1.10 一种连续神经隐函数 DeepSDF<sup>[57]</sup>的示意图。该图片引用自工作 DeepSDF 原文。我们可以在空间当中采样查询点，并带入连续神经隐函数查询符号距离值，从而获得零值面，进行网格的重建。

## ② 连续隐函数

相对于隐式的体素表示，近几年更加流行的三维几何表示是连续隐函数。和隐式的体素表示一样，连续隐函数同样没有直接把物体表面的点信息表征出来，

而是需要额外的表面提取过程。但是更进一步地，连续隐函数甚至连体素上的张量形式场数据都不需要，而是采用了连续函数来存储这些体信息。对于空间中的任何一个点，我们都可以通过在函数中代入查询坐标的方式获取其查询值。

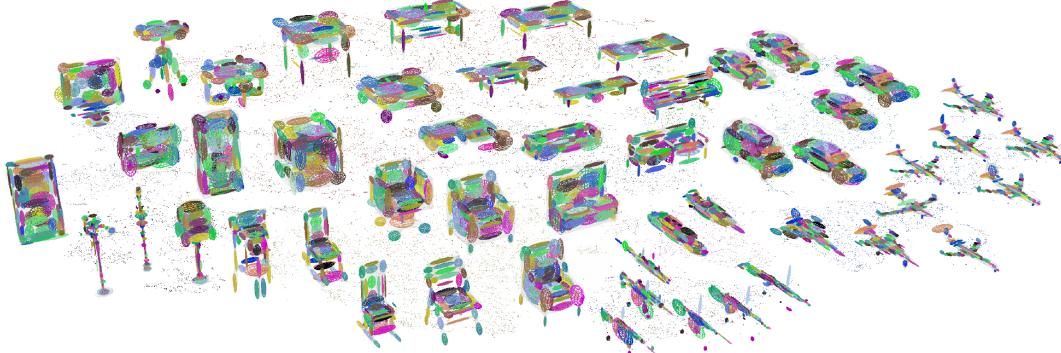


图 1.11 将 3D 形状表示为多个局部隐函数的组合的策略。图片引用自工作 LISF<sup>[58]</sup>。可以看到每个物体都被多个椭球的并集覆盖，这代表着多个局部隐函数的组合。

连续隐函数表示最近在一些应用中显示出优于离散表示（例如点云、网格和体素等）的巨大优势，因为它具有连续性并且不受有限分辨率的限制。它还在许多应用中取得了出色的性能，包括三维形状表示<sup>[57,59-60]</sup>，新视角合成<sup>[53-54]</sup>和多视角重建<sup>[61-62]</sup>等等。

对于 3D 形状表示，DeepSDF<sup>[57]</sup> 和 ONet<sup>[59]</sup> 分别提出使用神经隐函数来预测有符号距离和占用概率。作为示例，本文在图1.10中展示了 DeepSDF 的效果示意。然而，这些框架在表示复杂形状方面很弱。为了缓解这个问题，最近的一些工作专注于几何细节保持和局部形状的约简。IF-Nets<sup>[63]</sup> 提取了可学习的多尺度特征，对 3D 形状的局部和全局属性进行编码。后续的一些工作，如 LGCL<sup>[64]</sup>、PatchNets<sup>[65]</sup>、LISF<sup>[58]</sup> 等，将 3D 形状表示为多个局部隐函数的组合，其中每个局部隐函数负责形状的一部分。此类方法的样例如图1.11所示。LIGR<sup>[66]</sup>更是直接将整个大场景切分成了很多的小块来进行学习，如图1.12所示。由于局部特征学习的策略，这些方法比使用单个整体隐函数对形状进行编码的方法能表达更高质量的细节。在第二、三章中，本文所提出的两种表示均是基于局部的连续隐式表示。

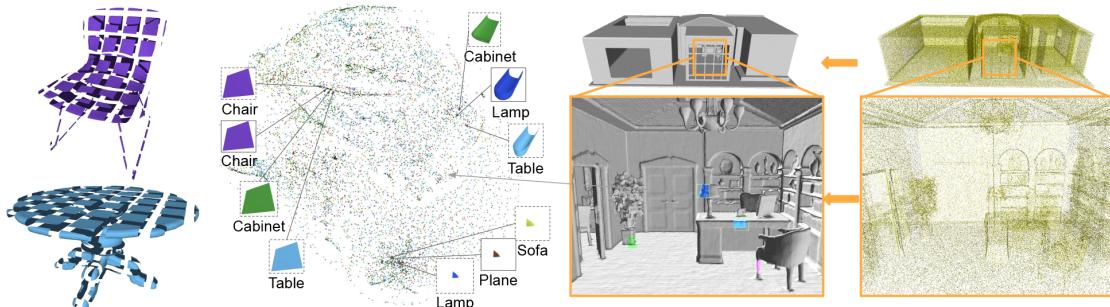


图 1.12 展示了一种局部格式的连续隐式表示。图片引用自工作 LIGR<sup>[66]</sup>。可以看到复杂的连续曲面被划分成了很多的小块，这降低了每一块的表示难度。

### 1.2.2 多物体间的几何形变表示

接下来的部分将分析多物体间的几何形变表示。对于单个的物体，其几何表示更多地把注意力集中在如何表示其形状；而对于多物体，其几何表示需要想办法来表征他们之间的运动匹配关系。本文探讨这种运动匹配关系的动机是：在很多情况下，我们需要将物体的运动进行跟踪与分析；还有一些情况下，我们需要在不同的时空视角下观测一个物体，然后将这些观测进行融合，组成一个全局的观测；甚至有的时候我们想要通过一个物体来驱动另一个物体，那么就要在它们之间建立关联。在上述的这些情况下都绕不开一个问题，就是如何对两个物体或者一个物体的两个状态进行运动或匹配的分析。

这个问题可以写成一个规范的数学形式。具体地，我们将其中一个模型记为源模型  $\mathcal{S}$ ，并将另一个模型记为目标模型  $\mathcal{T}$ 。为了构造它们之间的几何配准，我们的目标是寻找一个合理的形变  $\phi$ ，使得源模型在这个形变下的结果  $\phi(\mathcal{S})$ ，能够在某种度量下接近目标模型，即  $\|\phi(\mathcal{S}) - \mathcal{T}\|$  足够小，其中  $\|\cdot\|$  代表某种形状距离范数，比如倒角距离（Chamfer Distance, CD）。

物体之间的形变配准关系可以分为刚性和非刚性两种。顾名思义，刚性变换代表物体本身的刚度较大，柔性较小，从而形变只包含旋转和平移。相对于刚性变换的就是非刚性变换。非刚性的物体是有柔性的，它是可以自己产生形变的。这也就导致了非刚性变换比刚性变换有着更高的自由度，从而更难预测。但是非刚性的物体也面临着一些约束，换句话说，它不一定是完全柔性的。首先非刚性的物体有可能有局部刚性的性质，比方说人体本身是一个整体非刚性的物体，但是对于不可弯曲的局部来说，我们可以近似的认为它是局部刚性的；再就是对于同种类型的非刚性物体，它或许也有一些先验属性需要考虑，比如说人的手臂只能在手肘处弯曲，而不可能在大臂或者小臂的中间出现弯折的现象，这也是我们在预测形变的时候，需要纳入考虑的先验。

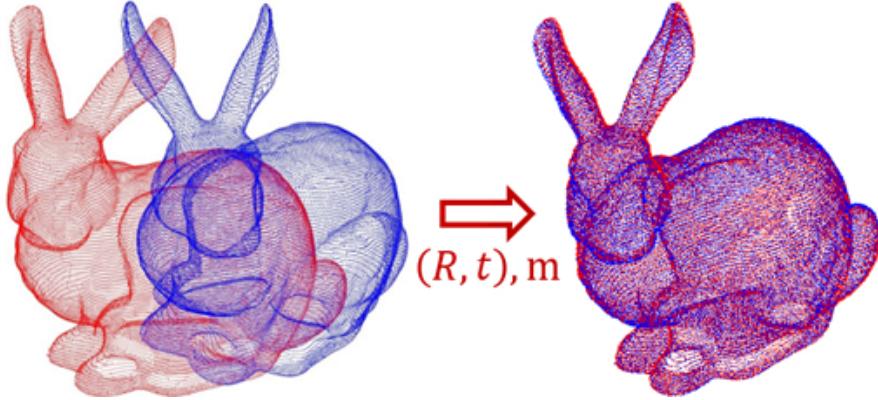
接下来本文将分别介绍刚性表示和非刚性表示的一些常见方法策略。刚性表示较为单纯；而非刚性表示却有着很多的设计方式，这些方式带给算法流程的运行难度和实验效果也有着很大的差别。

#### (1) 刚性表示

对于源模型  $\mathcal{S}$ ，其刚性变换可以写成如下形式：

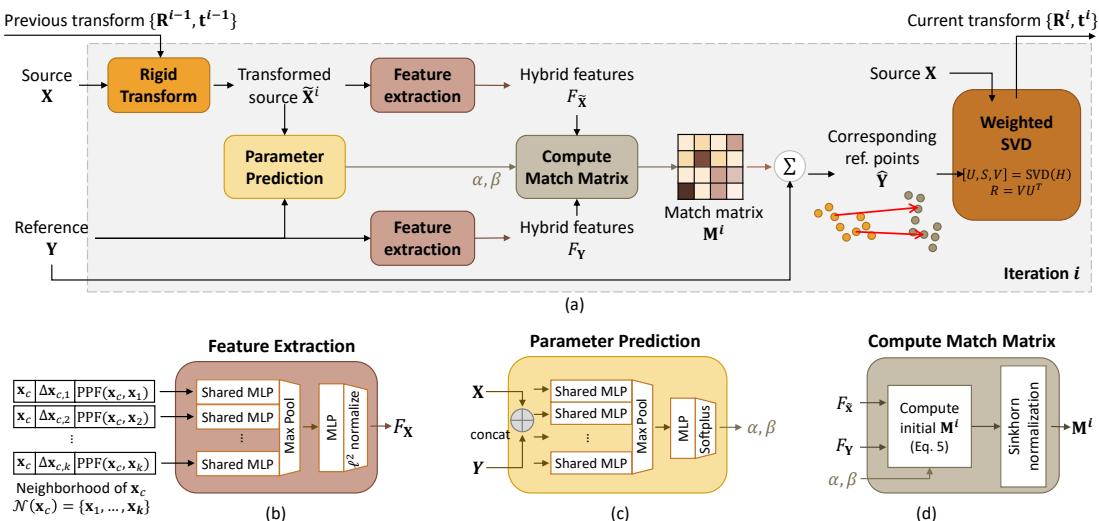
$$\phi(\mathcal{S}) = \mathbf{R} \cdot \mathcal{S} + \mathbf{t},$$

其中  $\mathbf{R}$  是旋转矩阵， $\mathbf{t}$  是平移向量。对于刚性变换，也有其他的表达格式，如四阶齐次矩阵、四元数组、李代数表示等，但是他们的表达能力与自由度都是和上式是相当的，且都可以在代数上进行等价的化归，本质上来说都表示了一个物体在不产生形变的情况下运动状况。

图 1.13 刚性配准过程示意图。该图片引用自工作<sup>[13]</sup>。

与刚性表示相关的任务是刚性配准问题，它旨在把两个具有潜在的刚性运动关系的物体进行注册，使其形状上接近，从而获得刚性的对齐效果。在数学上，即使得  $\|\mathbf{R} \cdot \mathcal{S} + \mathbf{t} - \mathcal{T}\|$  最小。为了直观的描述，本文在图1.13中展示了刚性配准的过程。

一种最经典的刚性配准方法是迭代最近点 (Iterative Closest Point, ICP)<sup>[67]</sup> 算法，该算法通过交替地搜索对应关系和估计刚性变换来解决刚性配准问题。在 ICP 之后，又出现了一些关于它的变体<sup>[68-71]</sup>，这些变体算法改善了其对噪声、异常值和不完整扫描的鲁棒性。另一方面，一些基于全局优化的方法<sup>[72-76]</sup>采取了不同的策略，这些方法搜索全局最优值，收敛会更稳定一点，但代价是计算速度慢。

图 1.14 刚性配准网络结构示例。该图片引用自 RPM-Net<sup>[77]</sup>。

最近，基于学习的方法也显示出很多可观的结果。3DMatch<sup>[78]</sup> 和 3DFeat-Net<sup>[79]</sup> 学习局部区域描述符而不是手工设计的特征来构建对应关系。Point-NetLK<sup>[80]</sup> 和 PCRNet<sup>[81]</sup> 提取输入点云的全局特征并迭代回归刚性变换。Deep Closest Point (DCP)<sup>[82]</sup> 改进了特征提取和对应关系预测阶段，通过 SVD 分解得到刚性变换。PR-Net<sup>[83]</sup> 和 RPM-Net<sup>[77]</sup> 提高了对异常值和部分可见性的鲁棒性。

最近, Feature-Metric<sup>[84]</sup> 提出了一种基于特征度量投影误差的半监督方法, 该方法也证明了对噪声、异常值和密度差异的鲁棒性。上述的很多方法都采用循环结构来迭代更新变换, 其中, 本文在图1.14中展示了一个刚性配准网络的示例。然而, 这些方法都仅仅适用于刚性变换, 而无法兼容非刚性变换。本文在第四章中提出一种新的循环迭代表示, 既可以适用于刚性变换, 又可以适用于非刚性变换, 而且可以非常方便地应用在神经网络之中。本文也会在该章节中展示所提出的表示对刚性变换的适用性和良好效果。

## (2) 非刚性表示

不同于刚性表示的单一表示方式、固定的自由度、较低的难度, 非刚性表示有很多不同的表示方式, 自由度更大, 且相应的非刚性配准任务难度也变得非常高。作为最直接的方式方法, 我们可以直接去对三维模型的每一个点都去估计一个形变偏移量, 但这样会造成过大的自由度和较高的学习难度。此外, 我们其实也没有必要去逐点估计一个形变偏移量, 因为即便物体是非刚性的, 它往往也会遵循某些约束, 比如局部刚性、保距性等。通过合理利用这些先验的约束, 可以使得我们用较低的自由度来表示出非刚性形变。本文在下面的部分会依次介绍如下几种非刚性表示: 逐点偏移场、样条函数、节点图。

### ① 逐点偏移场

首先是形式最简单的逐点偏移场方法。对于源模型  $S \in \mathbb{R}^{M \times 3}$ , 其通过逐点偏移场表示的非刚性变换可以写成如下形式;

$$\phi(S) = S + \Delta S,$$

其中  $\Delta S \in \mathbb{R}^{M \times 3}$  是逐点偏移量。

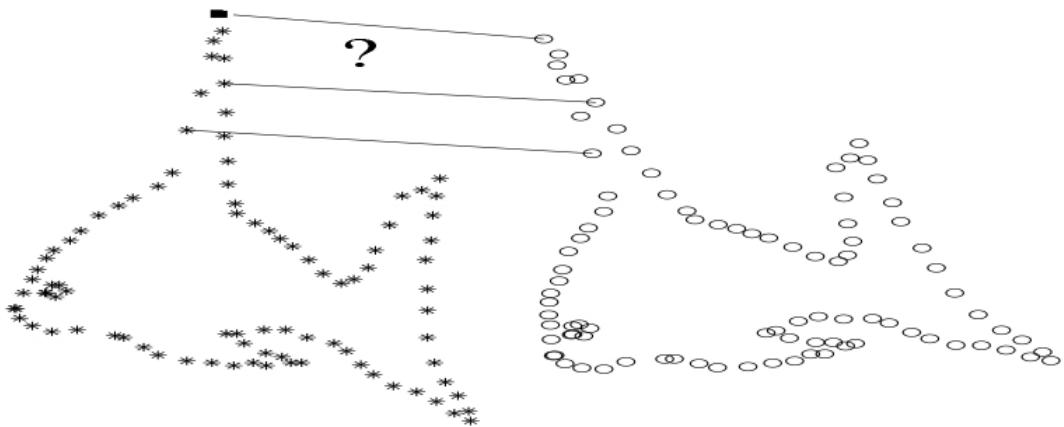


图 1.15 逐点偏移场的示意图。该图片引用自经典的非刚性配准工作 CPD<sup>[85]</sup>。

一些基于相干点漂移策略的方法<sup>[85-89]</sup>是最有代表性的以逐点偏移场作为非刚性表示的方法。在最经典的相干点漂移法(Coherent Point Drift, CPD)<sup>[85]</sup>之后, 又有了很多变式, 其中贝叶斯相干点漂移(Bayesian Coherent Point Drift, BCPD)<sup>[89]</sup>

是最近提出的贝叶斯版本的 CPD，在收敛性和有效性方面性能更好。本文在图1.15中展示了 CPD 方法使用的逐点偏移场格式。在逐点偏移的表达形式之上，这些方法使用了基于高斯混合模型（Gaussian Mixure Model, GMM）的策略构造能量函数，用源模型的点构造 GMM 概率密度函数，并通过期望最大化 (EM) 算法估计其变形，最大化目标曲面上的点的概率密度值。同样，近年的工作 CPD-Net<sup>[90]</sup> 使用了神经网络的方式来预测逐点偏移量，其算法结构如图1.16所示。此外，一些场景流估计工作<sup>[91-93]</sup>也采用了估计逐点位移的策略。

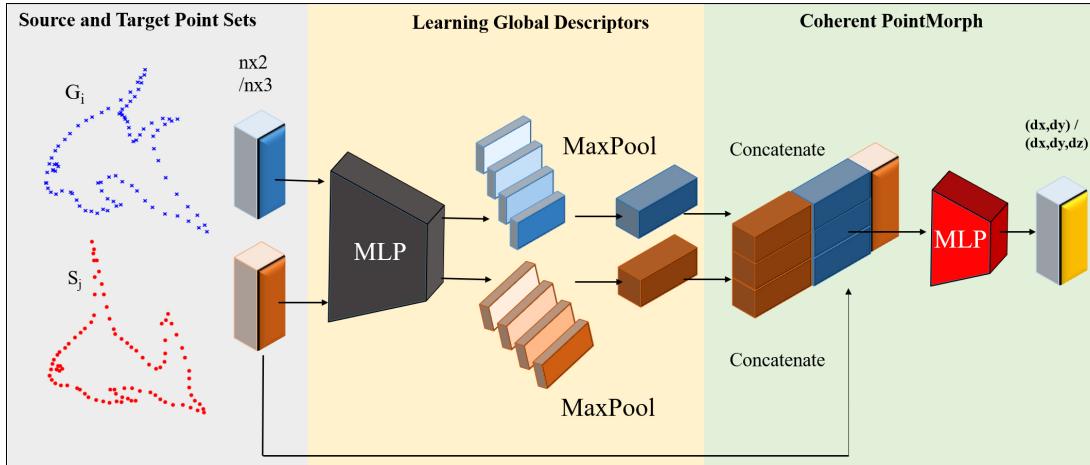


图 1.16 用神经网络预测逐点偏移场的网络示意图。该图片引用自 CPD-Net<sup>[90]</sup>。

正如之前讨论的，这种基于逐点偏移的方式虽然形式简单，但是却会造成过大的自由度和较高的学习难度。因此它只能处理较小的形变，当面临比较大的形变的时候可能会难以优化。本文在后续的章节中也会和基于这种表示的方法进行效果的对比。

## ② 样条函数

基于样条的方法相比于逐点偏移场，其形变自由度受到了很好的控制。此类方法的主要形式为，在空间选取一些控制点  $\{\mathbf{c}_i\}_{i=1}^I$ ，并采用某种与控制点有关的函数，不妨记为  $\psi(\cdot, \mathbf{c}_i)$ ，那么其形变表示可以一般地记为：

$$\phi(\mathbf{x}) = \sum_{i=1}^I w_i \cdot \psi(\mathbf{x}, \mathbf{c}_i),$$

也即，形变可以通过控制点上定义的一些基函数来加权获得。

经典的工作薄板样条函数非刚性点云注册方法（Thin Plate Spline Point Registration Method, TPS - RPM）<sup>[94]</sup>首次提出将薄板样条函数（Thin Plate Spline, TPS）用在非刚性注册工作中，其控制点选在了源模型上的点。之后有了很多变体<sup>[95-97]</sup>，均采用了薄板样条函数的方法。基于薄板样条的非刚性形变效果如图1.17所示。除去上面的基于优化的工作以外，基于学习的方法 PR-Net<sup>[98]</sup>将控制点选在了包围盒的体素点，并用神经网络预测用样条函数表示的形变信息。

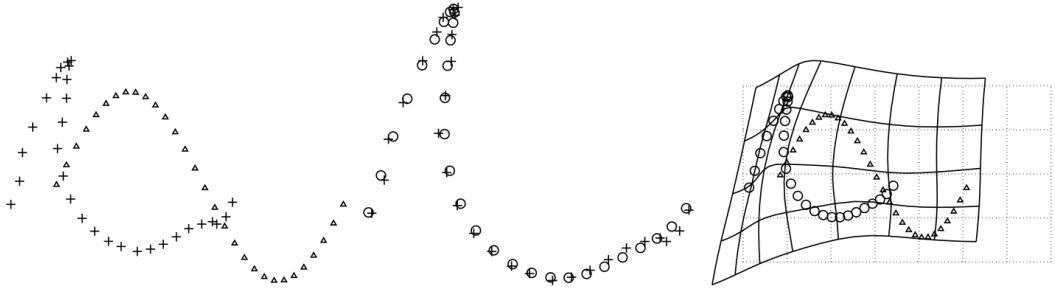


图 1.17 用薄板样条进行非刚性配准的示意图。该图片引用自 TPS-RPM<sup>[94]</sup>。左图是初始的源模型（三角形）和目标模型（加号形）；中间是形变结果（圆形）和目标模型；右图展示了一个规整的方格图案根据薄板样条进行形变的结果。

这种表示的自由度取决于控制点的选取。假如取了太多的控制点，那么自由度甚至要大过直接预测逐点偏移场；但是如果控制点太少，那么也许根本就没有足够的表达能力来表示想要表示的形变。除了点数多少，点的位置也很重要。假如控制点选的离模型曲面本身很远，那么这样的控制点起到的作用也会很小。这些限制使得它并不是我们在神经网络中预测非刚性形变时，首选想要的表示。本文在后续的章节中，也会和这种表示的方法进行对比。

### ③ 节点图

基于节点图的表示是另一种可以约束形变自由度的方法。节点图<sup>[99]</sup>将非刚性变形建模为一系列局部仿射变换，从而使得复杂的非刚性变换分解到了各个节点的局部仿射变换上。在源模型  $\mathcal{S}$  上定义图表示  $(\mathcal{V}, \mathcal{E}, \mathcal{M})$ ，其中：

$$\begin{aligned}\mathcal{V} &= \{\mathbf{v}_i \in \mathbb{R}^3, i = 1, \dots, m\}, \\ \mathcal{E} &= \{(\mathbf{v}_i, \mathbf{v}_j), \mathbf{v}_i \text{ 相邻于 } \mathbf{v}_j\}, \\ \mathcal{M} &= \{(R_i, t_i) \in SE(3), i = 1, \dots, m\},\end{aligned}\tag{1.3}$$

分别表示  $\mathcal{S}$  的节点集、边连接关系集和节点仿射变换集。通过在  $\mathcal{S}$  上定义运动  $\mathcal{M}$ ，源模型  $\mathcal{S}$  可以变形为  $\mathcal{M} \circ \mathcal{S}$ ，其公式形如：

$$\phi(\mathbf{x}) = \sum_{j=1}^J \alpha_j (R_j(\mathbf{x} - \mathbf{u}_j) + \mathbf{u}_j + \mathbf{t}_j),$$

其中  $\{\alpha_j\}_{j=1}^J$  为有关  $\mathbf{x}$  的  $J$  个测地邻居节点  $\{\mathbf{u}_j\}_{j=1}^J$  的系数，其详细具体计算方式本文将会在第五章中详细讨论。

图1.18展示了基于节点图的形变效果。在之前的工作中，基于节点图进行非刚性配准的方法<sup>[100-101]</sup>达到了很好的效果，其效果优于逐点偏移场和基于样条的表示，这得益于其很好地利用了源模型的几何结构先验。但是其在深度学习上的使用并不是一件自然的事情，具有一定的设计障碍，并且之前的基于节点图的方法都是优化方法而非基于深度学习的方法。而基于优化的方法，相比于数据驱动的深度学习方法而言是脆弱的，比较容易陷入局部最优，从而在某些样例上失

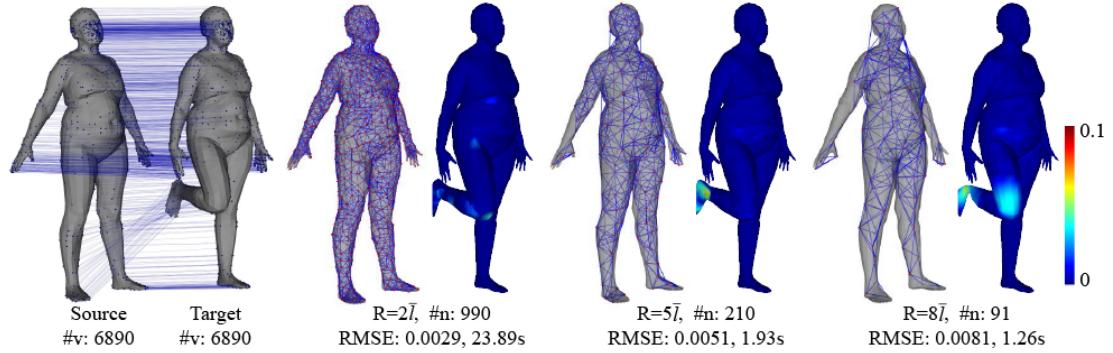


图 1.18 基于节点图的形变示意图。该图片引用自工作<sup>[100]</sup>。可以看到可形变模型在节点图的驱动下进行形变，展示出了节点图良好的表达能力。

败。本文在后续的章节中将会提出一种基于可微形变节点图的非刚性形变表示，把基于传统优化的节点图流程搬到深度学习框架来解决非刚性配准问题，从而结合了节点图表示和深度学习的优势，取得了更好的效果。

### 1.3 研究内容与结构安排

本文主要研究三维几何表示及其在相应的三维视觉任务当中的应用。关于三维几何表示本文将其分为两种大的类型：单个物体的几何形状表示，以及多物体间的三维形变表示。针对单个物体的几何形状表示，本文首先是提出了一种专门针对点云的局部连续隐式表示；又分析了在超高精度要求下的三维几何表示的需求，并提出了一种基于径向基函数的局部形状表示。对于多物体间的形变表示，本文首先是提出了一种基于逐点加权刚性形变的非刚性几何形变表示，并将其用在了循环神经网络的框架当中；之后为了进一步的提高神经网络的效果，本文在神经网络中加入结构化的几何先验，设计了一种基于可微形变节点图的非刚性形变表示，进一步提高了神经网络处理非刚性配准的能力。

在后续的章节中，本文的结构安排如下。本文将在第二章中介绍基于局部神经场的点云连续隐式表示，并展示其在点云上采样任务上的优良效果。之后，在第三章当中，本文将介绍所提出的基于径向基函数的局部形状表示，并将其使用到三维模型的扫描细节修复任务上。在第四章当中，本文将详细描述基于逐点加权刚性形变的非刚性几何形变表示，并展示出这种表示使用在神经网络当中的强大能力。紧接着，在第五章当中，本文将介绍基于可微形变节点图的非刚性形变表示，并展示这种表示在保持神经网络端到端可训练的前提下，可以进一步提升非刚性配准的效果。最后，第六章将对全文进行总结，并畅想未来有可能的研究方向。

## 第2章 基于神经场的点云局部隐式表示

### 2.1 引言

正如前文介绍，点云是三维场景最基本和最流行的表示形式，且已广泛应用于三维重建<sup>[16,33,55,102]</sup>，虚拟/增强现实<sup>[103-104]</sup>和自动驾驶<sup>[4-6]</sup>等许多应用中。在传统的点云表示中，每个点只表示三维空间中的一个位置，如果分配其法向量，也仅是可以进一步扩展为表示局部平面。因此，点云的表达能力会受到其分辨率的限制。尽管在之前的工作中已经有人提出了点云上采样方法<sup>[10-11,105-106]</sup>来从低分辨率获得高分辨率点云，这相当于变相提高了点云的表达能力，但它们的策略仍然是“离散到离散”的方式，无法克服当前点云表示的分辨率限制。

在本章节的工作中，本文提出了神经点云（Neural Points），一种新颖的点云表示，它显著提高了表达能力。与传统表示不同，Neural Points 的每个点都编码一个通过神经场表示的局部曲面小块。具体来说，每个逐点曲面小块都表示为二维参数域和三维局部曲面小块之间的局部同构，并且同构通过神经场隐式表示。由于其连续性，神经场使 Neural Points 具有优于传统离散点云表示的几个优势，包括不受有限分辨率的限制，对复杂几何形状的强大表达能力。同时，所有局部曲面小块共享训练好的神经场模型，因此 Neural Points 的存储开销很小。

与现有的一些通过一个神经场<sup>[57,62]</sup> 表示整个曲面模型的方法不同，传入 Neural Points 的神经场和点特征都是局部的。通过这种设计，本文的 Neural Points 表示显现出几个优势，包括出色的细节表达能力、强大的泛化能力和对训练数据的低需求。具体来说，本文基于二维流形上的三维局部小块与二维简单参数圆盘同构的事实，使用了局部神经场来构建二维参数域和三维局部曲面小块之间的连续双射。使用神经场，每个三维局部小块都可以被视为在二维参数域中定义的参数曲面。为此，本文采用编码器来提取逐点局部特征，并将局部特征作为传递到神经场网络的输入的一部分。有了局部形状信息，神经场可以很好地表示曲面块，训练好的网络模型被所有三维模型的所有局部块共享。最后，本文设计了一种聚合策略，将所有局部神经场聚合在一起，形成最终的全局形状。Neural Points 模型采用高分辨率且几何细节丰富的几何曲面训练，训练后的 Neural Points 模型可以表示多种多样的几何细节。

作为 Neural Points 的自然应用，本文基于提出的 Neural Points 表示进行点云上采样任务。给定一个低分辨率的点云，点云上采样的目标是预测具有更高分辨率的上采样点云，以便更好地捕捉潜在的连续曲面的几何特征。现有的点云上采样方法<sup>[10-11,105-106]</sup>通常采用“离散到离散”的方式，每个训练好的模型只对一个上采样因子起作用。与这些方法不同，本文将输入的离散点云表示转换为连续

的 Neural Points 表示，然后可以对任意目标分辨率进行相应的采样。

实验结果表明，本文提出的 Neural Points 可以产生高质量的连续曲面，并很好地克服了点分辨率的限制。对于点云上采样任务，本文的 Neural Points 表示比传统的点云表示实现了更好的性能。总之，这项工作的贡献包括：

- 本文提出 Neural Points，一种新颖的点云表示，其表达能力大大优于传统的点云表示，存储开销低。
- 本文使用神经隐函数和曲面块的深层局部特征来表示局部神经场，并设计一种聚合策略来形成最终的全局形状。
- 评估结果表明，Neural Points 表示对各种输入具有出色的鲁棒性和泛化能力。将 Neural Points 应用于点云上采样任务比现有方法实现了更好的性能。

## 2.2 基于神经场的点云局部隐式表示：设计与应用

### 2.2.1 概述：基于点云的表示

点云  $\mathcal{X} = \{\mathbf{x}_i \in \mathbb{R}^3\}_{i=1}^I$  是其潜在连续曲面  $\mathcal{S}$  的离散表示。在传统的点云表示中，每个点只表示一个三维位置，它的表达能力完全取决于它的分辨率。提高其表达能力的一种直接策略是进行点云上采样：

$$\mathcal{S} \xrightarrow{\text{Discretize}} \{\mathbf{x}_i\}_{i=1}^I \xrightarrow{\text{Upsample}} \{\mathbf{x}_i^r\}_{i,r=1}^{I,R} \subset \mathcal{S}, \quad (2.1)$$

其在之前的点云上采样工作中也被一直研究着<sup>[11,106]</sup>。然而，公式(2.1)中的上采样方式虽然可以一定程度上提高点云的分辨率，但是依然是离散到离散的。由于其上采样结果仍然是离散的，所以本质上依然受分辨率限制。

在本章节的工作中，本文提出了 Neural Points，一种新颖的点云表示，它比传统的点云具有更好的表达能力。Neural Points 表示采用离散到连续的策略，这与公式(2.1)的形式完全不同。给定输入点云  $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^I$ ，本文将潜在的连续曲面描述为：

$$\mathcal{S} \xrightarrow{\text{Discretize}} \{\mathbf{x}_i\}_{i=1}^I \xrightarrow{\text{Neural Points}} \mathcal{S}' \approx \mathcal{S}, \quad (2.2)$$

其中  $\mathcal{S}'$  是由 Neural Points 表示的连续曲面。通过公式(2.2)中的形式，本文提出的框架可以克服点云分辨率的限制，并在  $\mathcal{S}'$  上实现任意因子的点云采样。在下文中，本文将描述所提出的 Neural Points 表示的算法细节。

### 2.2.2 算法流程

整个算法流程在图2.1 中给出。(1) 给定输入点云，本文首先为每个局部小块构建局部神经场，这是基于局部参数化的；(2) 将局部神经场聚合在一起形成全

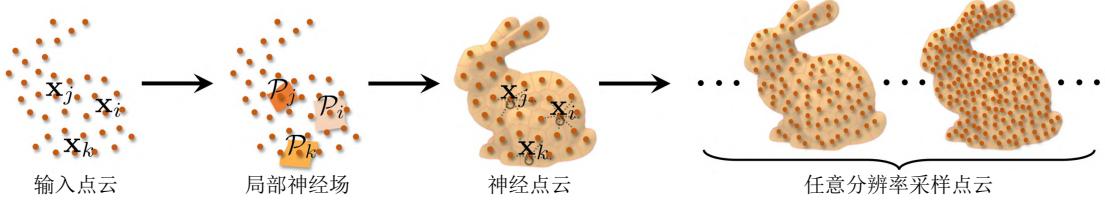


图 2.1 基于神经场的点云局部隐式表示。对于输入点云，通过局部连续神经场表示离散的逐点局部块，通过聚合所有局部神经场构建全局连续 Neural Points 曲面。通过在构建的连续 Neural Points 曲面上采样，可以生成任意分辨率的点云。

局隐式曲面；(3) 通过构建的连续神经表示，可以重新采样任意数量的点。下面，本文将介绍每个组件的详细信息。

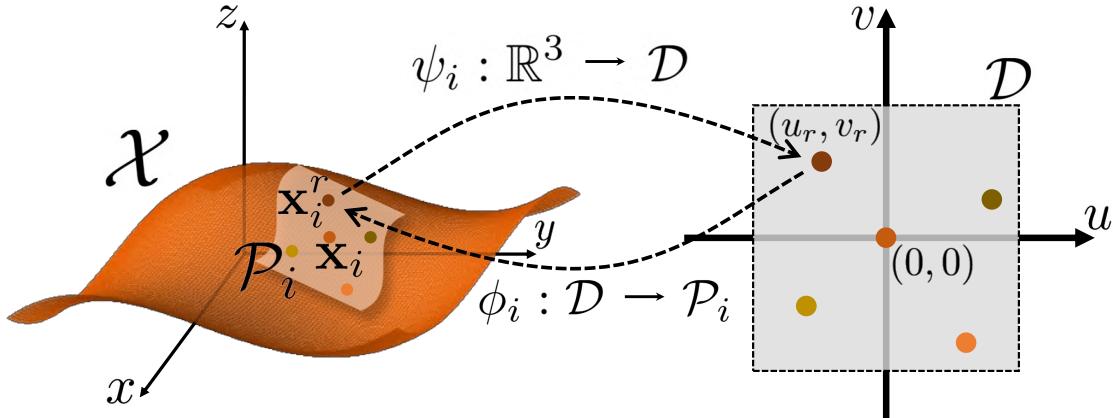


图 2.2 局部神经场，以及同构的三维小块与二维采样域之间的双射映射函数。 $\mathcal{P}_i$  是围绕中心点  $\mathbf{x}_i$  的三维局部曲面片， $\mathcal{D}$  是二维参数域。 $\phi_i$  和  $\psi_i$  是构建  $(u_r, v_r) \in \mathcal{D}$  和  $\mathbf{x}_i^r \in \mathcal{P}_i$  之间对应关系的映射。

**局部神经场。**本文使用局部神经场来描述潜在的连续曲面。输入点云仍然表示为  $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^I$ 。就像在<sup>[107]</sup>中一样，本文以  $\{\mathbf{x}_i\}_{i=1}^I$  为中心点将曲面划分为重叠的局部块  $\{\mathcal{P}_i\}_{i=1}^I \subset \mathbb{R}^3$ 。在每个点  $\mathbf{x}_i$ ，三维局部小块  $\mathcal{P}_i$  同构于二维参数域  $\mathcal{D} \subset \mathbb{R}^2$  (本文中使用  $\mathcal{D} = [-1, 1]^2$ )，这意味着可以在它们之间构造一个双射映射：

$$\phi_i : \mathcal{D} \rightarrow \mathcal{P}_i, \quad \psi_i : \mathbb{R}^3 \rightarrow \mathcal{D},$$

其中  $\psi_i|_{\mathcal{P}_i} = \phi_i^{-1}$ 。本文在图2.2中展示了神经场的细节。给定任意二维采样点  $(u_r, v_r) \in \mathcal{D}$ ，可以计算  $\mathbf{x}_i^r = \phi_i(u_r, v_r) \in \mathcal{P}_i$  作为其对应的三维采样点。类似地，对于任何三维点  $\mathbf{x}_i^r \in \mathcal{P}_i$ ，本文可以计算其对应的二维坐标  $(u_r, v_r) = \psi_i(\mathbf{x}_i^r) \in \mathcal{D}$ 。此外，本文可以通过以下方式计算点  $\mathbf{x}_i^r$  处的法向：  $\mathbf{n}_{\mathbf{x}_i^r}$ ：

$$\mathbf{n}_{\mathbf{x}_i^r} = \nabla_u \phi_i \times \nabla_v \phi_i,$$

然后将其标准化为单位长度。

**神经场聚合。**虽然局部神经场  $\{\phi_i\}_{i=1}^I$  可以构造二维参数域和三维局部小块之间的对应关系，但不同的局部神经场是在不同的局部坐标系中定义的。因此，本文需要将它们聚合成一个全局连续神经场的曲面。

对于  $\forall \mathbf{x} \in \mathbb{R}^3$ , 本文的目标是将  $\mathbf{x}$  拉到基于神经场的曲面上。本文在  $\{\mathbf{x}_i\}_{i=1}^I$  到  $\mathbf{x}$  中找到最近的中心点, 并将邻居点的索引集表示为  $\mathcal{N}(\mathbf{x})$ 。对于  $\forall k \in \mathcal{N}(\mathbf{x})$ , 根据  $\mathbf{x}$  到  $\mathbf{x}_k$  的距离计算组合权重:

$$w_k = e^{-\alpha_1 \|\mathbf{x} - \mathbf{x}_k\|_2^2}, \forall k \in \mathcal{N}(\mathbf{x}). \quad (2.3)$$

使用  $w_k$ , 基于神经场的曲面上的点可以通过以下方式计算:

$$\rho(\mathbf{x}) = (\sum_{k \in \mathcal{N}(\mathbf{x})} w_k \cdot \phi_k \circ \psi_k(\mathbf{x})) / (\sum_{k \in \mathcal{N}(\mathbf{x})} w_k), \forall \mathbf{x} \in \mathbb{R}^3. \quad (2.4)$$

具体来说, 输入点  $\mathbf{x}$  通过邻居神经场映射到二维参数域, 变为  $\psi_k(\mathbf{x}) \in \mathcal{D}$ , 然后将二维点映射回到三维, 坐标为  $\phi_k \circ \psi_k(\mathbf{x}) \in \mathcal{P}_k$ 。这样, 就构建了一个基于神经场的连续曲面。

点  $\rho(\mathbf{x})$  的法向计算如下:

$$\mathbf{n}_{\rho(\mathbf{x})} = (\sum_{k \in \mathcal{N}(\mathbf{x})} w_k \cdot \mathbf{n}_{\phi_k \circ \psi_k(\mathbf{x})}) / (\sum_{k \in \mathcal{N}(\mathbf{x})} w_k), \forall \mathbf{x} \in \mathbb{R}^3, \quad (2.5)$$

并将组合向量归一化为单位长度。注意  $\{\mathbf{n}_{\phi_k \circ \psi_k(\mathbf{x})}\}_{k \in \mathcal{N}(\mathbf{x})}$  的方向, 本文需要在计算加权和之前将其调整为沿相同方向。

**点云采样。**本文还可以从表示连续曲面的神经场中重新采样点云。考虑到点云采样过程本质上是应用于二维流形, 本文在二维参数域  $\mathcal{D}$  中开始采样操作。

本文在二维参数域  $\mathcal{D}$  中均匀地采样点, 并将二维采样点映射到三维局部块上。具体来说, 对于中心点  $\mathbf{x}_i$ , 本文统一采样  $R$  个点  $\{(u_r, v_r) \in \mathcal{D}\}_{r=1}^R$  和然后将它们映射到三维变为  $\{\mathbf{x}_i^r \in \mathcal{P}_i\}_{r=1}^R$ 。对于整个输入点云, 本文从所有小块中获得采样点的并集:

$$\mathcal{X}_R = \bigcup_{i=1}^I (\{\mathbf{x}_i^r \in \mathcal{P}_i\}_{r=1}^R) = \{\mathbf{x}_i^r \in \mathcal{P}_i\}_{i,r=1}^{I,R}.$$

然后本文可以从  $\mathcal{X}_R$  中统一采样  $J$  个点:

$$\mathcal{Y}^* = \{\mathbf{y}_j^* \in \bigcup_{i=1}^I \mathcal{P}_i\}_{j=1}^J,$$

其中  $J$  的值在训练和推理阶段是任意的。然后本文用公式(2.4)将  $\mathbf{y}_j^*$  拉到 Neural Points 曲面上:

$$\mathcal{Y} = \{\mathbf{y}_j = \rho(\mathbf{y}_j^*)\}_{j=1}^J.$$

### 2.2.3 网络结构

这一部分将介绍网络结构, 包括局部特征提取和局部神经场。

**局部特征提取。**如上所述，对于每个中心点  $\mathbf{x}_i$ ，本文提取一个局部特征作为输入到神经场  $\phi_i$  的一部分，本文将其表示为  $\mathbf{c}_i$ 。对于每个  $\mathbf{x}_i$ ，本文首先提取其相邻点  $\{\mathbf{x}_k\}_{k \in \mathcal{N}(\mathbf{x}_i)}$  并去中心化到  $\{\mathbf{x}_k - \mathbf{x}_i\}_{k \in \mathcal{N}(\mathbf{x}_i)}$ 。然后本文应用 DGCNN<sup>[19]</sup> 主干提取具有  $|\mathcal{N}(\mathbf{x}_i)|$  个点的点集的特征。具体来说，本文使用了几个 EdgeConv 层，将每一层的特征聚合起来，然后传递到另一个 EdgeConv 层和 max-pooling 层得到  $\mathbf{c}_i^*$ 。对于每个  $i$ ，本文将  $\mathbf{c}_i^*$  与其邻居的局部池化聚合以获得最终的局部特征：

$$\mathbf{c}_i = \mathbf{c}_i^* \oplus \text{MaxPool}\{\mathbf{c}_k^*\}_{k \in \mathcal{N}(\mathbf{x}_i)},$$

其中  $\oplus$  表示聚合操作。

**神经场网络结构。**本文使用配备 ReLU 激活层的基于 MLP 的网络来表示  $\phi_i$ 。隐函数由所有三维模型的所有小块共享，它们表示为  $\Phi$ 。正如本文上面提到的， $\Phi$  的输入应该包含  $\mathbf{c}_i$  所涉及的局部形状信息和作为查询点的二维参数坐标。

对于传入  $\phi_i$  的二维查询坐标，本文借鉴之前的工作<sup>[54]</sup>中的位置编码设计，并将位置编码函数表示为  $\gamma$ 。位置编码与局部特征聚合，作为神经隐函数的输入。具体来说，可以将  $\Phi$  表示为：

$$\Phi((u_r, v_r), \mathbf{c}_i, \mathbf{x}_i) = \mathbf{x}_i + \theta_\Phi(\gamma(u_r, v_r) \oplus \mathbf{c}_i),$$

其中  $\theta_\Phi$  是  $\Phi$  中的 MLP。自然地，对于每个  $i$ ，可以写出公式  $\phi_i(\cdot) = \Phi(\cdot, \mathbf{c}_i, \mathbf{x}_i)$ 。

另一个关键函数是  $\psi_i$ ，它是  $\phi_i^{-1}$  的延拓。在本文的实现中， $\psi_i$  定义为：

$$\psi_i(\mathbf{x}) = \phi_i^{-1}(\text{Proj}(\mathbf{x}, \phi_i(\mathcal{D}))), \forall \mathbf{x} \in \mathbb{R}^3, \quad (2.6)$$

其中本文使用  $\text{Proj}(\mathbf{x}, \phi_i(\mathcal{D}))$  来表示  $\phi_i(\mathcal{D})$  中到  $\mathbf{x}$  的最近点。这个定义的好处是本文可以将  $\phi_i \circ \psi_i$  表示为：

$$\phi_i \circ \psi_i(\mathbf{x}) = \text{Proj}(\mathbf{x}, \phi_i(\mathcal{D})), \forall \mathbf{x} \in \mathbb{R}^3, \quad (2.7)$$

在公式(2.4) 中使用。

最后，本文在公式(2.6) 和公式(2.7) 中解释了本文的投影过程。在本文的实现中，使用  $\phi_i(\mathcal{D})$  的近似值：

$$\phi_i(\mathcal{D}) \approx \{\mathbf{x}_i^r\}_{r=1}^R.$$

然后，应该从三维点和点集制定投影操作。为了描述方便，考虑一个三维点  $\mathbf{p}$  和一个点集  $\mathcal{Q} = \{\mathbf{q}_t\}_{t=1}^T$ 。在  $\mathcal{Q}$  中找到  $\mathbf{p}$  的相邻点的索引集，并将其表示为  $\mathcal{N}(\mathbf{p}; \mathcal{Q})$ 。从  $\mathbf{q}$  到  $\mathcal{Q}$  的投影公式为：

$$\text{Proj}(\mathbf{p}, \mathcal{Q}) = (\sum_{k \in \mathcal{N}(\mathbf{p}; \mathcal{Q})} w_k \cdot \mathbf{q}_k) / (\sum_{k \in \mathcal{N}(\mathbf{p}; \mathcal{Q})} w_k), \quad (2.8)$$

其中  $w_k$  计算为:

$$w_k = e^{-\alpha_2 \|\mathbf{p} - \mathbf{q}_k\|_2^2}, \forall k \in \mathcal{N}(\mathbf{p}; \mathcal{Q}). \quad (2.9)$$

与公式(2.5)类似, 可以得到其法向  $\mathbf{n}_{Proj(\mathbf{p}, \mathcal{Q})}$ 。

### 2.2.4 损失函数

在这一部分中, 本文将介绍损失函数项, 包括对曲面形状、点法向和聚合质量的约束。总损失为:

$$\mathcal{L} = \mathcal{L}_{shape} + \omega_1 \cdot \mathcal{L}_{nor} + \omega_2 \cdot \mathcal{L}_{int}.$$

在本文当前的实现中, 具有更高分辨率的点云被用作真实值, 其他形状表示也可以相应地使用。本文将真实值点云表示为  $\mathcal{Z} = \{\mathbf{z}_j\}_{j=1}^L$  并将  $\mathbf{z}_l$  的法向表示为  $\mathbf{n}_{\mathbf{z}_l}$ 。如 2.2.2 所述, 框架的输出可以总结为  $\mathcal{X}_R = \{\mathbf{x}_i^r; \mathbf{n}_{\mathbf{x}_i^r}\}_{i,r=1}^{I,R}$  和  $\mathcal{Y} = \{\mathbf{y}_j; \mathbf{n}_{\mathbf{y}_j}\}_{j=1}^J$ 。

监督是离散的, 但  $\mathcal{X}_R$  和  $\mathcal{Y}$  应该被视为来自连续曲面的任意采样。为了用离散监督来监督连续曲面, 本文没有使用基于最近点搜索的 CD 损失, 尽管它在之前的点云上采样工作中被广泛使用<sup>[106,108]</sup>。相反, 本文采用公式(2.8) 中描述的投影策略。一般来说, 对于两个点云  $\mathcal{P} = \{\mathbf{p}_s\}_{s=1}^S$  和  $\mathcal{Q} = \{\mathbf{q}_t\}_{t=1}^T$ , 本文将它们的距离定义为:

$$d(\mathcal{P}, \mathcal{Q}) = \frac{1}{S} \sum_{s=1}^S \|\mathbf{p}_s - Proj(\mathbf{p}_s, \mathcal{Q})\|_2^2.$$

进一步地, 将它们的逐点法向差异定义为:

$$d_{\mathbf{n}}(\mathcal{P}, \mathcal{Q}) = \frac{1}{S} \sum_{s=1}^S \|\mathbf{n}_{\mathbf{p}_s} - \mathbf{n}_{Proj(\mathbf{p}_s, \mathcal{Q})}\|_2^2.$$

为了约束  $\mathcal{X}_R$  和  $\mathcal{Y}$  接近  $\mathcal{Z}$ , 损失项为:

$$\mathcal{L}_{shape} = d(\mathcal{X}_R, \mathcal{Z}) + d(\mathcal{Z}, \mathcal{X}_R) + d(\mathcal{Y}, \mathcal{Z}) + d(\mathcal{Z}, \mathcal{Y}).$$

为了监督逐点法向, 损失项是:

$$\mathcal{L}_{nor} = d_{\mathbf{n}}(\mathcal{X}_R, \mathcal{Z}) + d_{\mathbf{n}}(\mathcal{Z}, \mathcal{X}_R) + d_{\mathbf{n}}(\mathcal{Y}, \mathcal{Z}) + d_{\mathbf{n}}(\mathcal{Z}, \mathcal{Y}).$$

此外, 本文还对聚合质量使用了损失项。来自相邻神经场的重叠形状应该相同, 并且构建的全局 Neural Points 曲面应该覆盖输入曲面。基于这些要求, 本文将损失项设计为:

$$\mathcal{L}_{int} = \sum_{j=1}^J \sum_{k \in \mathcal{N}(\mathbf{y}_j; \mathcal{X})} \|\mathbf{y}_j - Proj(\mathbf{y}_j, \{\mathbf{x}_k^r\}_{r=1}^R)\|_2^2.$$

## 2.3 实验结果和讨论

本节将给出实现细节、消融实验、结果、比较以及泛化和鲁棒性测试。

### 2.3.1 实现细节

**数据集。**本文在 PUGeo-Net<sup>[107]</sup> 收集的 Sketchfab<sup>[109]</sup> 数据集上训练和测试本文的模型，其中包含 90 个训练模型和 13 个具有丰富几何特征的测试模型。本文使用相同的数据集训练所有比较方法以进行公平比较。与其他点云上采样方法类似，本文采用泊松圆盘采样<sup>[110]</sup> 算法在模型上提取点，以获得上采样算法的输入和真实值。具体来说，本文提取 10000 个点作为整个输入点云，并提取 40000 和 160000 个点作为整个真实值点云，用于上采样 4 倍和 16 倍的实验。本文还按照之前的点云上采样工作来提取一些锚点和相邻的局部部分（整个点云的子集）作为网络的输入，而不是将整个模型传递到网络中。总之，本文在每个训练模型上选择 1000 个锚点，在每个测试模型上选择 114 个锚点。在训练过程中，输入点云的点数设置为 256。

为了测试本文训练的模型的有效性，本文不仅在 Sketchfab 的测试集上进行测试，而且还在更多没有见过的数据集上进行测试，而无需重新训练本文的网络。本文在 PU-GAN<sup>[105]</sup> 收集的数据集上测试所有方法，其中测试模型的数量为 27。该数据集中测试模型的形状相对简单，因此本文在每个模型上提取 2000 的点作为整个输入点云。除了上面提到的合成数据外，本文还对真实捕获的数据进行了进一步的测试。本文还在 iPhone X 上的深度传感器捕获的点云上测试并评估了本文的方法。

**实验设置。**传入网络的所有输入都归一化到三维单位球内。本文在所有的实验中设置  $R = [4 \cdot J/I]$ 。在局部特征提取网络中，相邻点的数量为 10。对于所有  $Proj(.,.)$  操作，本文将相邻点的数量设置为 4。对于损失项， $\omega_1$  和  $\omega_2$  分别设置为 0.01 和 0.3。公式(2.3) 中的  $\alpha_1$  和公式(2.9) 中的  $\alpha_2$  分别设置为  $10^2$  和  $10^3$  以计算指数权重。本文在 DGCNN 主干中使用了 5 个卷积层，在 MLP 中使用了 3 个线性层。本文将批大小设置为 8，并以 25000 的迭代次数训练网络。学习率从 0.01 开始，每 1250 次迭代乘以 0.5。对于所有比较方法，本文使用与本文自己的方法相同的设置来训练它们。该模型使用 PyTorch<sup>[111]</sup> 进行训练。所有训练和测试均在具有四个 32G V100 GPU、32 个 Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz 和 128GB RAM 的工作站上进行。本文经过训练的网络模型可以应用于所有点云。Neural Points 的存储开销是存储预训练模型，其总大小仅为 2.53Mb（局部特征提取网络为 1.35Mb，神经场 MLP 网络为 1.18Mb）。

**评估指标。**类似于最近的点云上采样工作<sup>[106,108]</sup>，本文采用倒角距离（Chamfer Distance, CD）、豪斯多夫距离（Hausdorff Distance, HD）和点到面（Point-to-Face

Distance, P2F) 作为指标。对于所有指标, 指标越小, 结果的质量就越好。

消融设定	CD	HD	P2F
去掉输入局部 KNN	2.49	9.54	3.03
去掉 $\mathcal{L}_{\text{nor}}$	0.70	3.46	0.75
去掉 $\mathcal{L}_{\text{int}}$	0.77	4.09	0.81
去掉聚合过程	0.83	5.64	0.88
整个模型	<b>0.66</b>	<b>3.32</b>	<b>0.69</b>

表 2.1 消融实验的结果, 指标为 CD( $\times 10^{-5}$ )、HD( $\times 10^{-3}$ ) 和 P2F( $\times 10^{-3}$ )。

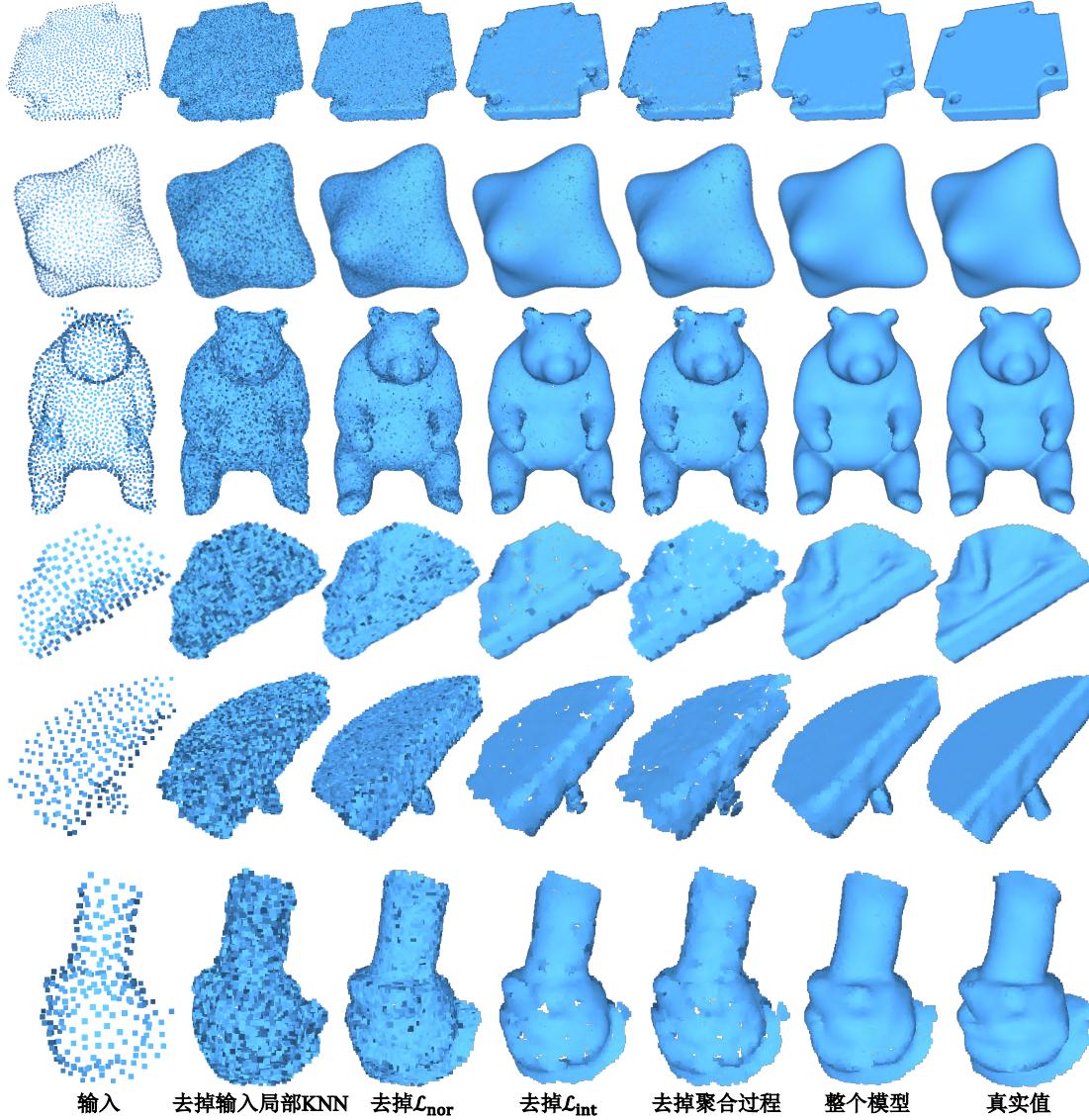


图 2.3 本文的消融实验的可视化结果。

### 2.3.2 消融实验

接下来, 本文进行消融实验以展示每个组件如何影响最终结果。具体来说, 本文使用 Sketchfab 数据集作为基准。本文主要针对局部特征提取器结构、损失项和聚合过程设计消融实验。消融实验的定量结果见表2.1。在第一行的实验中,

本文为整个输入点云应用了 DGCNN 主干，而不是局部 KNN 点集。可以看到，定量结果很差，这验证了有必要为 KNN 点集使用共享主干网络，并限制特征提取器中感受野的大小。第 2-3 行展示了不同的损失项设置，其中分别删除了  $\mathcal{L}_{\text{nor}}$  和  $\mathcal{L}_{\text{int}}$  项。可以看到，删除这两项中的任何一项都会使结果不如完整模型。其中，去掉  $\mathcal{L}_{\text{int}}$  损失项会大大降低算法性能。最后，本文去掉聚合过程和所有相关的算法设置，并在第 4 行展示结果。可以看到，没有聚合过程的结果比完整模型差得多。

图2.3展示了消融实验的可视化结果。如果对整个点云而不是局部 KNN 点集应用主干网络，训练效果会很差，视觉结果看起来很乱。如果去掉法向的损失项  $\mathcal{L}_{\text{nor}}$ ，训练效果将不如完整模型，结果看起来不够光滑。如果删除聚合损失项  $\mathcal{L}_{\text{int}}$ ，结果将包含一些空洞。此外，当去掉所有与聚合相关的过程时，结果不仅有空洞，而且还会出现一些拼接位错。与这些算法设置相比，本文的完整模型可以产生令人满意的结果。

采样因子	方法	Sketchfab			PU-GAN		
		CD	HD	P2F	CD	HD	P2F
4x	PU-Net	5.93	4.98	4.51	23.61	13.91	10.02
	PU-GAN	3.30	3.45	3.61	16.79	9.36	7.04
	PU-GCN	2.85	3.21	2.79	14.74	11.97	6.36
	Dis-PU	2.61	3.25	2.67	13.79	11.83	7.14
	PUGeo-Net	2.28	<b>2.10</b>	1.04	11.26	3.54	2.14
	本文方法	<b>2.17</b>	2.46	<b>0.93</b>	<b>8.17</b>	<b>3.08</b>	<b>1.59</b>
16x	PU-Net	5.25	5.82	5.99	20.70	15.49	12.16
	PU-GAN	2.97	3.99	3.76	11.89	10.81	7.48
	PU-GCN	2.35	3.84	3.02	11.37	12.69	6.95
	Dis-PU	2.36	3.79	3.31	12.75	13.65	8.09
	PUGeo-Net	0.83	3.50	0.97	3.58	7.14	1.94
	本文方法	<b>0.66</b>	<b>3.32</b>	<b>0.69</b>	<b>3.35</b>	<b>6.52</b>	<b>1.49</b>

表 2.2 4x 和 16x 上采样的结果和比较，指标为 CD( $\times 10^{-5}$ )、HD( $\times 10^{-3}$ ) 和 P2F( $\times 10^{-3}$ )。

### 2.3.3 结果和比较

本文将本文的方法与之前的点云上采样方法进行比较：PU-Net<sup>[11]</sup>、PU-GAN<sup>[105]</sup>、PU-GCN<sup>[108]</sup>、Dis-PU<sup>[106]</sup> 和 PUGeo-Net<sup>[107]</sup>。其中，PUGeo-net、PU-GCN 和 Dis-PU 是最近提出的实现最先进性能的方法。所有比较方法都是离散到离散的，并且使用与本文的方法相同的训练集进行训练。对于所有方法，本文分别训练两个版本：4x 和 16x。在每个版本中，本文使用相应分辨率的真实值数据训练所有方法。为了公平比较，本文在所有方法之间共享相同的实验设置（包括批大小、迭代次数、优化器、学习率等）。

4x 和 16x 的统计结果和对比见表2.2 和图2.4。可以看到，本文的方法在数量和质量上都达到了最佳性能。PU-Net 的结果总体上是混乱的。PU-GAN 比 PU-

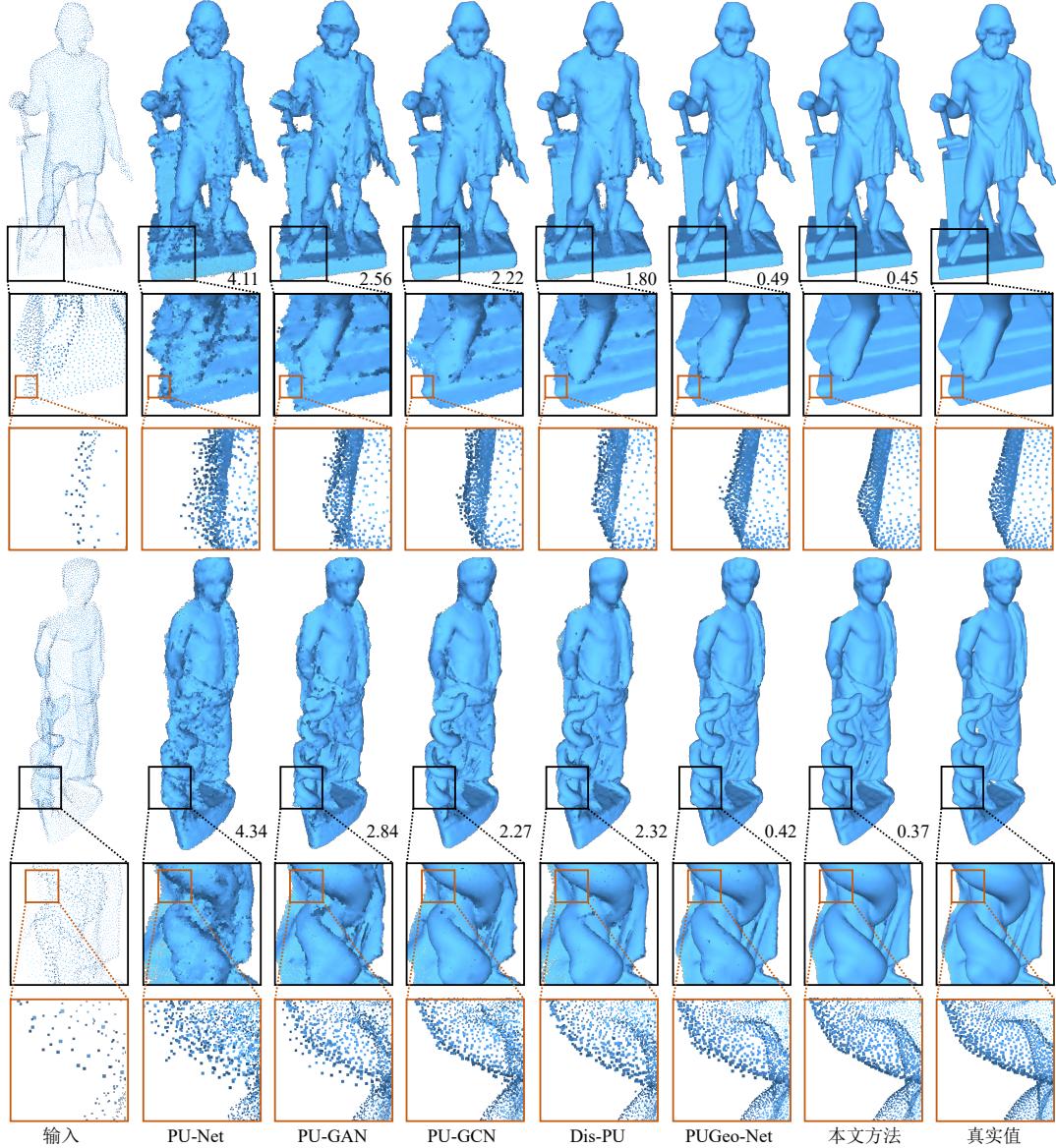


图 2.4 Sketchfab<sup>[109]</sup> 数据集上的结果和比较。误差度量  $CD (\times 10^{-5})$  也在底部给出。为了更好的可视化，本文放大结果的一些局部部分并选择适当的视角来展示细节。

Net 表现更好，但也会产生一些奇怪的噪声和异常值。PU-GCN 的性能优于上述两种方法，可以保留平坦区域，但仍会在特征丰富的区域产生一些噪声点。本文尽最大努力微调 Dis-PU<sup>[106]</sup> 的训练设置。在平坦区域上，Dis-PU 表现非常好，这得益于其解耦的细化方案。然而，在高曲率区域，Dis-PU 无法产生高质量的结果。在本文的实验中，PUGeo-Net 取得了比所有其他比较方法更好的结果。然而，从放大的角度来看，可以发现其上采样点的分布并不像本文的结果那么平滑。这是因为 PUGeo-Net 以离散和独立的方式对每个小块进行上采样。从结果中，可以观察到本文的方法生成的结果是全局平滑的，并且包含丰富的局部几何细节。

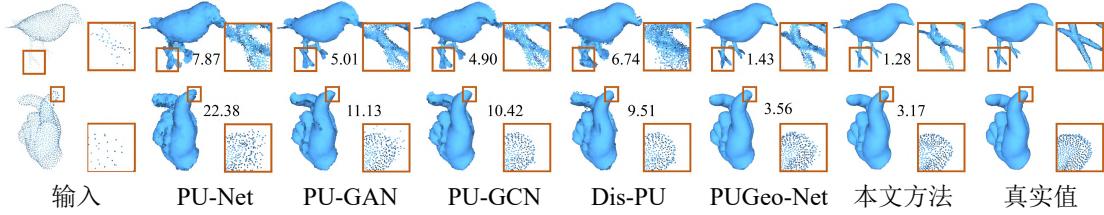


图 2.5 PU-GAN<sup>[105]</sup> 数据集上的结果和比较。误差度量  $CD (\times 10^{-5})$  也在底部给出。为了更好的可视化，本文展示了一些局部部分以便更好地进行比较。

### 2.3.4 泛化能力和鲁棒性

**对未见数据的泛化。**本文在之前工作中常用的另一个数据集上进行测试，即 PU-GAN<sup>[105]</sup> 收集的数据集。在不重新训练的情况下，本文再次将训练后的模型应用于 PU-GAN 数据集并展示定性和定量结果。结果和比较在表2.2 和图2.5 中。可以看到，本文的模型在定量结果和可视化结果上仍然取得了最好的表现，证明了 Neural Points 表示的良好泛化能力。

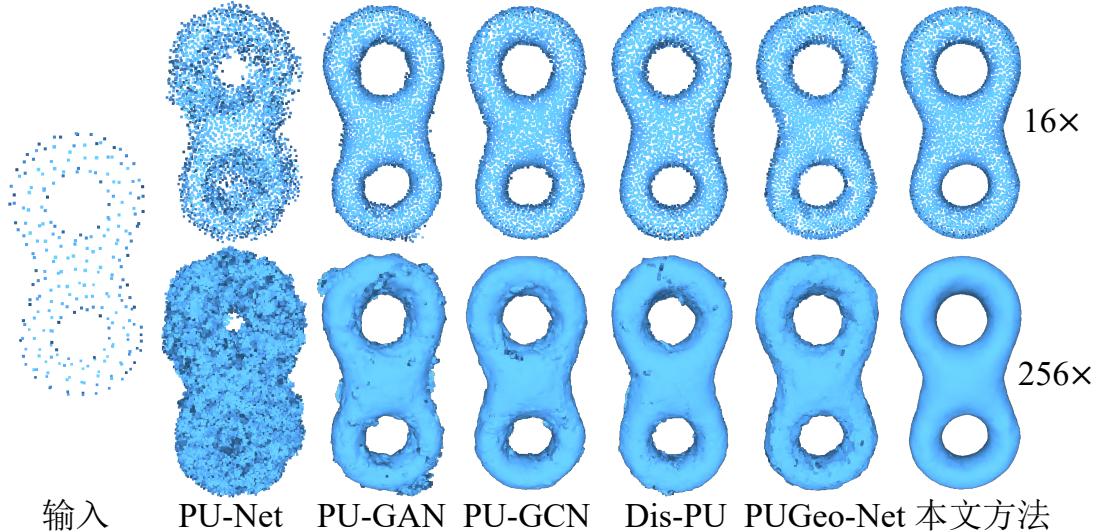


图 2.6 在非常大的因子下的上采样结果。

**对较大采样因子的鲁棒性。**本文探讨了具有非常大的上采样因子的不同方法的上采样效果。请注意，包括本文自己的方法在内的所有测试方法都是使用  $16\times$  监督信号进行训练的。本文将方法中的  $N$  调整为  $256 \cdot I$  以获得  $256\times$  结果。对于其他方法，本文将训练好的  $16\times$  模型应用两次，以获得  $256\times$  的结果。结果如图2.6所示。不同方法进行  $16\times$  上采样的可视化结果并没有太大的差异，而  $256\times$  上采样的结果则相差很大。模型应用两次后，PU-GAN、PU-GCN、Dis-PU 和 PUGeo-Net 的结果出现了一些易于观察的缺陷。同时，本文的结果仍然保持良好的质量。由于 Neural Points 以连续且不受分辨率限制的方式表示曲面，因此本文的结果不会受到采样因子大小的影响。

**对真实捕捉数据的鲁棒性。**本文进一步将本文的方法与两种最具代表性的方法

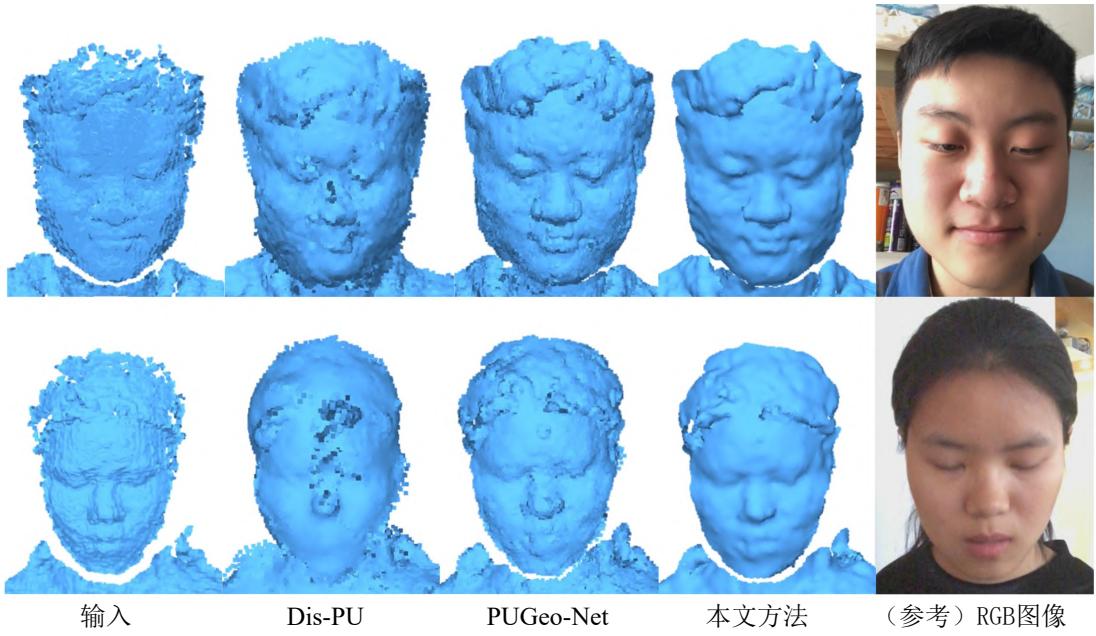


图 2.7 在深度相机拍摄的点云上的对比。展示 RGB 图像是为了更好地可视化，本文的方法中没有使用这些图像。

(Dis-PU<sup>[106]</sup> 和 PUGeo-Net<sup>[107]</sup>) 在实际捕获的数据上进行比较。具体来说，本文使用 iPhone X 上配备的深度传感器捕获人脸的深度图像。以深度图像转换的点云作为输入，本文采用不同的方法对点云进行上采样。如图2.7 所示，本文的方法即使是在扫描点云包含很多噪声和凹凸不平区域的情况下依然表现非常好。相比之下，Dis-PU<sup>[106]</sup> 生成的结果未能保留几何特征，并包含一些缺陷。PUGeo-Net<sup>[107]</sup> 的结果保留了输入点云的一些几何特征，但仍然可以在其结果中观察到一些缺陷。尤其需要注意的是，实际扫描数据具有可变的输入顶点数、非均匀等问题，在此情况下，本文方法依然获得了良好的效果。该测试验证了 Neural Points 对实际捕获数据的鲁棒性和有效性。

### 2.3.5 更多结果和分析

**神经场小块的可视化。**为了清楚地展示每个点的神经场，本文在图2.8 中可视化了从神经场生成的一些局部小块。上图在第一行展示了本文提取局部特征的局部点云，然后在第二行展示了它们对应的局部神经场小块。中心点呈现为红色。为了清楚地展示局部潜在形状，上图在第一行中展示了本文从中提取局部点的潜在曲面，在第二行中展示了放大的局部部分上生成的神经场小块。可以看到，生成的小块可以平滑地覆盖局部区域，并且与局部点云的形状非常吻合。每列展示从本文测试集中的模型中提取的不同局部部分。对于不同的局部形状，神经场可以获得满意的局部块，这对 Neural Points 表示的整体表达能力有很大贡献。

**具有更多采样因子的结果。**如论文所述，本文训练的模型可以应用于任意采样因子。本节展示了具有更多采样因子的结果，并且本文的模型也支持非整数因

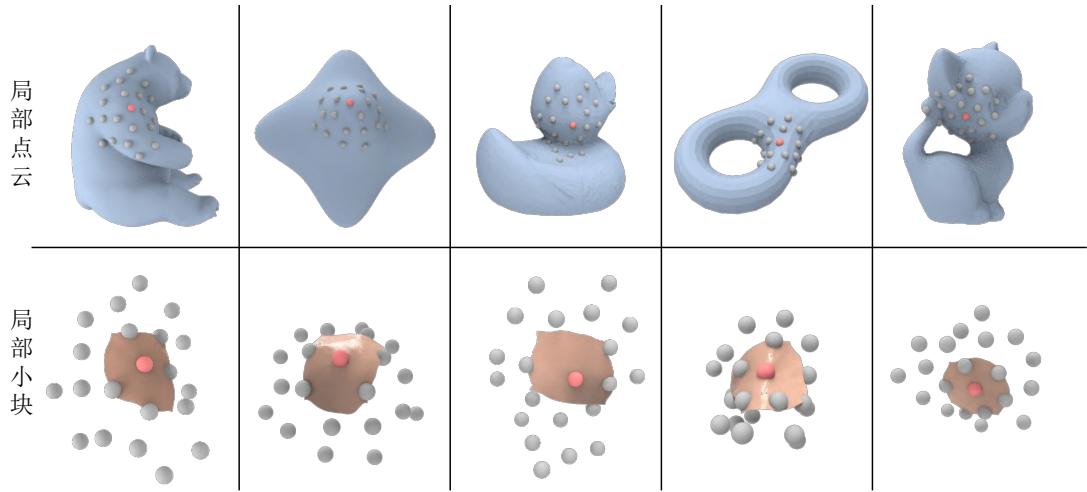


图 2.8 本文的 Neural Points 表示的局部小块的可视化。上图第一行展示了一些局部点云的例子，其中中心点是红色的。为了清楚地展示潜在的局部形状，本文还展示了从中提取局部点的潜在曲面。利用本文算法中从神经场生成的相应局部小块展示在第二行。每列展示测试集中不同的局部部分。

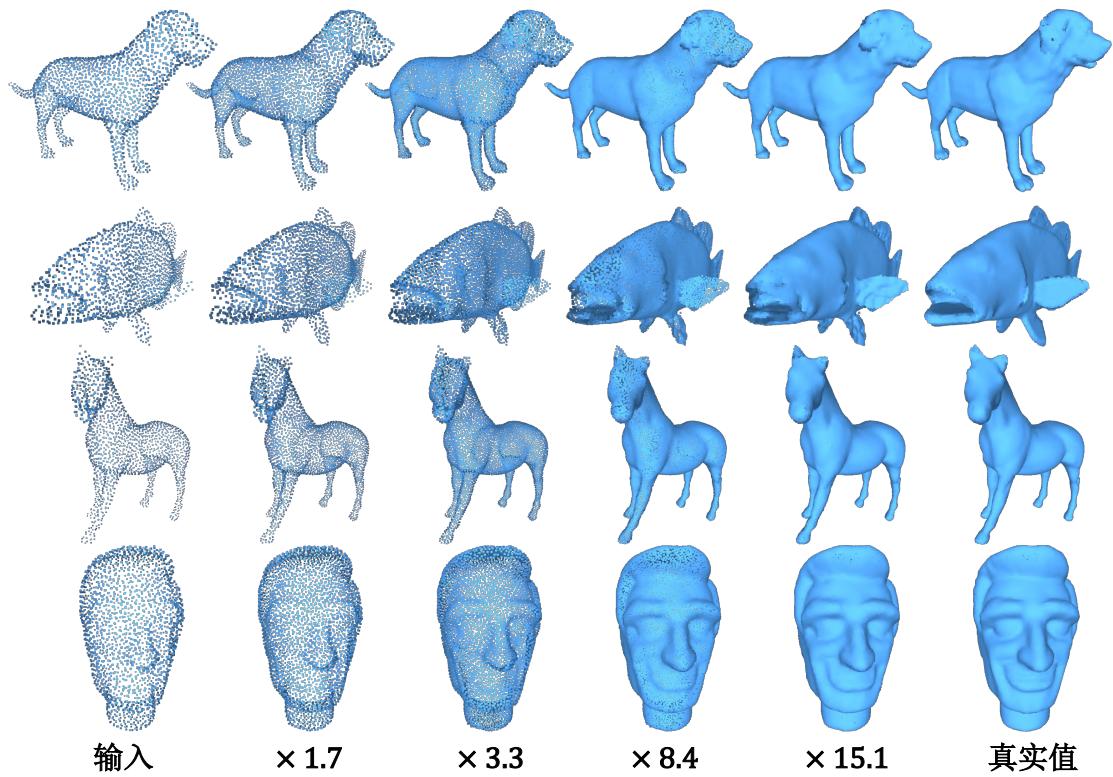


图 2.9 在任意采样因子下的结果，包括非整数因子。

子。在图2.9中，本文展示了一些示例。具体来说，本文使用因子1.7、3.3、8.4和15.1对输入点进行上采样。可以看到，本文的模型可以在大范围内对各种采样因子产生良好的结果。由于篇幅有限，本文只展示了这些采样因子，但本文的模型也支持其他采样因子。

## 2.4 本章小结

本文提出了一种基于神经场的点云局部隐式表示，称为 Neural Points，其中每个点通过神经场表示局部连续几何形状，而不仅仅是三维空间中的位置或局部平面。Neural Points 可以表达更复杂的细节，因此比传统的点云具有更强的表达能力。本文用包含丰富几何细节的曲面训练 Neural Points，使训练的模型对各种形状都有足够的表达能力。具体来说，本文提取了点上的深层局部特征，并通过二维参数域和三维局部块之间的局部同构来构建神经场。通过聚合神经场获得最终的全局连续曲面。Neural Points 强大的表达能力、鲁棒性和泛化能力已通过大量实验得到验证。将其应用于点云上采样任务的出色性能进一步验证了其良好的特性。

## 第3章 基于径向基函数的局部形状表示

### 3.1 引言



图 3.1 本章节提出一种新的几何表示，并展示其在三维模型扫描细节恢复任务上的效果。一个完整的几何细节恢复框架如图所示。(a) 中展示了扫描真实物体并恢复高质量细节的完整过程。为了清楚地展示模型如何从输入变为输出，本文可视化逐点偏移，展示从高质量网格到输入粗糙网格的有符号距离。黄色区域表示位移向外，紫色区域表示位移向内。在平坦区域，低频特征得到了很好的保留。在细节丰富的高频区域，高频特征得到了很好的恢复。(b) 中展示了一些放大的局部部分。

在诸如三维重建<sup>[16,33,55,102,112-113]</sup>等三维视觉重要任务中，我们有时会遇到模型的复杂度非常高的情况，这种情况对三维几何表示的要求自然也会很高。为了适应对模型复杂度、几何精度要求很高的任务情形，一种好的三维几何表示应该具有充分的拟合能力来拟合模型的复杂细节，再就是这种表示需要适合在深度学习的框架中使用。与此同时，良好的泛化能力可以让模型方便跨场景迁移；对于拓扑变化的表达能力可以使我们不受物体拓扑的限制。这些同样需要被视作我们选取或设计三维几何表示的重要判据。

传统的表示往往没有办法在这几方面做到兼容。基于网格的深度学习方法<sup>[38]</sup>很难处理输出拓扑不固定的情况；点云<sup>[18-19]</sup>和体素<sup>[114]</sup>均可以处理输出拓扑不固定的情况，但是毕竟是离散的表示，用其表示高精度模型会非常占用内存和计算资源。连续隐式表示<sup>[57,62]</sup>往往可以做到突破分辨率的限制，但是进一步地，即便如此，其泛化能力还是没有办法得到保证，使得这些方法很多时候只能在单个物体或单个数据集上训练。在三维模型的复杂度较高的时候，这些表示或多或少有其局限性，这给所有牵涉到高精度几何建模的问题带来了挑战，也激励了本文提出一种表示来解决上述挑战。

为了在一个具体的场景下分析问题，本文首先分析在本文中所考虑的具体任务：三维扫描模型的几何细节修复。高质量三维网格重建是计算机图形学和计算机视觉领域的热门研究问题，在文化遗产重建和逆向工程等领域有着广泛的应用。如今，低成本扫描设备（如 *Kinect*<sup>[7]</sup>）的发展为人们提供了更多使用

三维扫描和重建的机会。尽管三维重建已被广泛研究多年<sup>[16,33,55,102,112-113]</sup>，但在某些情况下恢复具有丰富几何细节的高质量网格仍然具有挑战性。具体来说，由于三维扫描设备分辨率和扫描物体大小的限制，捕捉细节丰富的高分辨率点云并非易事。此外，点云重建技术（例如 Screened Poisson<sup>[16]</sup>）也会导致细节丢失，尤其是在低扫描分辨率下。在某些情况下，我们还需要使用早期公共数据集中获得的低质量扫描数据<sup>[115-116]</sup>。硬件设备限制和原始数据质量都促使我们寻求一种更好的方法，该方法能够恢复在扫描物体期间丢失的几何细节并生成能够很好地描述潜在物体曲面的高质量网格。

与数字几何处理领域的经典问题（如网格简化<sup>[24-25]</sup>和网格滤波<sup>[26-27]</sup>）相比，三维扫描恢复主题很少被研究。扫描仪超分辨率工作<sup>[117]</sup>提出对一个物体进行多次扫描，取所有扫描的平均值，以获得更好的扫描效果。扫描点云超分辨率工作<sup>[118]</sup>通过聚合扫描点云上相似的小块来对单视角扫描点云做超分辨率。此外，点云上采样<sup>[10-11,119]</sup>也可以看作是用来解决细节恢复问题的一种可选途径。点云上采样过程通过用更多的点表示潜在曲面来恢复几何特征，但它仍然需要从上采样的点云中重建网格，这并非易事，并且在此过程中可能会丢失几何特征。另一种相关技术是网格细分<sup>[22-23,120-122]</sup>。虽然网格细分会生成具有更多顶点的网格，但不能保证保留或增强几何特征。网格细节增强<sup>[27,123-125]</sup>直接对网格上的几何特征进行操作。然而，依靠这种方法获得的细节改进完全取决于一系列手工设定的参数，这在实际应用中是不稳定的。

三维网格表示的不规则性使问题变得具有挑战性。由于该任务是不稳定的，因此需要先验几何知识。考虑到基于学习的方法已广泛应用于许多几何处理任务，如分类<sup>[126-127]</sup>、重建<sup>[16,33,55,113]</sup>，深度神经网络是我们问题的自然解决方案。然而，由于三维网格表示的不规整性，将输入直接传递到网络中并非易事。为了解决这个挑战，本文不直接使用网格表示并预测欧几里得顶点坐标，而是采用了基于连续隐式表示<sup>[58-59,66,128]</sup>的策略，使用可微且规整的三维数据形式，从而在本文的问题中允许结果的拓扑发生变化。在本文的工作中，本文选择连续隐式表示，径向基函数（Radial Basis Function，RBF）<sup>[129]</sup>，并在 RBF 参数空间中执行操作，从而实现细节恢复效果。

该问题也因三维物体的高复杂性而变得具有挑战性。对于 RBF 来说，为了很好地拟合整个物体，必须使用高维 RBF 参数，这是相当具有挑战性的。幸运的是，分治策略可以在一定程度上缓解困难。一个关键的观察是：扫描过程中引起的细节误差是局部的，与全局结构关系不大。此外，在整个物体的不同局部位置，引起细节扫描误差的原因是类似的。因此，不同局部部分的细节误差在小规模的局部小块上，是共享相似的分布的，并且可以通过共享的网络进行推断。具体来说，本文用 RBF 拟合每个局部小块，并对每个小块而不是整个物体执行细节恢

复操作。本文将提出的网络称为几何细节恢复网络 (Geometric Detail Recovering Network, GDR-Net)，并将在后续的章节中介绍其网络结构。

综合以上两点考虑，本文归纳所使用的三维几何表示为一种基于径向基函数的局部形状表示，并解决了当前任务中所面临的上述两点挑战。至于如何将隐式表示转换为显式网格，之前的工作始终依赖于行进立方体 (Marching Cubes) [37] 算法的许多变体之一，而该算法不是完全可微的。在本文的工作中，本文使用神经网络而不是不可微的后处理来重建网格小块。对网格小块，本文使用了一种可微的三维几何表示，可微行进立方体场 (Deep Marching Cubes Layers, DMCL) [17]。在测试阶段，本文将所有小块的 DMCL 融合起来，从而形成一个完整的水紧的网格。

为了训练网络，本文构建了一个由合成模型和真实扫描模型共同组成的数据集，包括高/低质量的网格对。实验结果表明，本文的所提出的基于径向基函数的局部形状表示，配合本文提出的名为 GDR-Net 的几何细节恢复网络，比现有方法实现了更好的细节恢复性能。本文还通过对各种类别的物体、各种设备下的输入、极低质量下的输入进行测试，通过定性和定量的实验结果，展示出本文所提出方法的出色的泛化能力。

总之，这项工作的贡献包括：

- 本文提出一种基于径向基函数的局部形状表示，具有很强的表达能力，并且允许隐空间上的操作。
- 本文将所提出的几何表示用在几何细节恢复任务上，利用误差分布的局部性，采用分治策略，来恢复扫描三维物体过程中丢失的细节，并生成可以很好地描述底层物体的高质量网格。
- 本文通过实验证明了该表示具有很强的泛化能力，可以在不重新训练的情况下直接应用到未见过的新物体。
- 本文构建了一个合成和真实扫描模型的数据集，包括高/低质量对，可供后续的学术研究。

## 3.2 基于径向基函数的局部形状表示：设计与应用

### 3.2.1 基于径向基函数的局部形状表示

给定一个粗糙的输入网格  $M_{in}$ ，本文的目标是生成一个细节恢复的网格  $M_{out}$ ，它比  $M_{in}$  更接近潜在的物体曲面  $M_{gt}$ 。像之前的工作 [66, 128] 所使用的隐式函数一样，本文通过目标域  $\Omega$  中的隐式函数  $f$  来表示三维曲面  $M$ ，其符合以下公式：

$$f(\mathbf{x}) = SDF(\mathbf{x}, M), \forall \mathbf{x} \in \Omega, \quad (3.1)$$

其中  $SDF(\mathbf{x}, M)$  表示从查询点到目标模型的有符号距离。在本文的工作中，本文用形状基函数的线性组合拟合  $f(\mathbf{x})$ :

$$f(\mathbf{x}) \approx \sum_{i=1}^N \lambda_i \phi(|\mathbf{x} - \mathbf{c}_i|), \forall \mathbf{x} \in \Omega, \quad (3.2)$$

其中  $\phi(|\mathbf{x} - \mathbf{c}_i|)$  是不同形状模型共享的形状基， $\mathbf{c}_i$  是形状基函数的中心， $N$  是基函数的数量。对于输入和输出曲面，本文使用同一组参数基来定义。基于这种表示，输入曲面和输出曲面的差别在于，它们具有不同的形状参数  $\{\lambda_i\}$ 。那么通过对形状参数进行修复，模型的几何就可以得到修复。

在本文中，对于形状基函数  $\phi(|\mathbf{x} - \mathbf{c}_i|)$  的具体选择，本文选择了 RBF。之所以选择 RBF，因为它展示了良好的曲面重建中的表达能力<sup>[129]</sup>。当然，其他形状基，如多项式<sup>[130]</sup>、样条曲线<sup>[131]</sup> 和有理样条曲线<sup>[132]</sup>等也都可以以相同的方式来进行使用。如果通过公式(3.2)中的公式来表示整个物体曲面，则形状基  $N$  的大小需要相当大，才能达到可接受的表示能力，这使得回归整个输出模型非常具有挑战性。幸运的是，本文可以使用分治的策略。具体地，本文将整个模型分成小块，并用 RBF 拟合每个小块，即将小块转化为 RBF 表示。对于每个小块，其拟合效果是比较容易做到令人满意的，同时本文并不需要将基函数  $N$  的数量设置的太大。综上所述，本文将几何细节恢复任务进行了转化，将其看作是对每个小块的 RBF 系数偏移量的回归。

### 3.2.2 算法流程

图3.2展示了本文的算法流程，包括三个步骤：分解 ((1))，逐片细节恢复 ((2)) 和网格融合 ((3))。第一步，本文将输入模型  $M_{in}$  划分为许多小的曲面小块，并通过 RBF 拟合每个小块。第二步是逐片使用粗糙的 RBF 参数回归恢复后的 RBF 参数并将结果解码为恢复后的网格小块，对每个曲面小块执行细节恢复过程。最后一步是通过组合所有网格小块来生成整个输出网格。在本章节中，为了方便描述，本文分别使用下标 “ $_{in}$ ”、“ $_{out}$ ” 和 “ $_{gt}$ ” 来表示输入、输出和真实值数据。下面，本文介绍每个部分。

#### (1) 分解

本文在离散体素格子的帮助下分解输入网格  $M_{in}$ 。输入网格  $M_{in}$  首先被归一化并放入其包围盒  $[0, 1]^3$ 。通过将边界框的长度、宽度和高度分成  $n$  部分，本文得到  $n \times n \times n$  个单元格和  $(n + 1) \times (n + 1) \times (n + 1)$  单元格节点，其中单元格代表最小的立方体，单元格节点代表单元格的角位置。为了提取小块，本文选择具有  $k^3$  个单元的块，其中心  $2^3$  个单元与  $M_{in}$  相交，并提取所选块与  $M_{in}$  的交叉部分。在选择训练块时，本文将步长设置为 1。在将已训练的模型应用于具体的样例时，本文将所选块之间的重叠限制在 8 以下，同时保持输入网格完全被所选

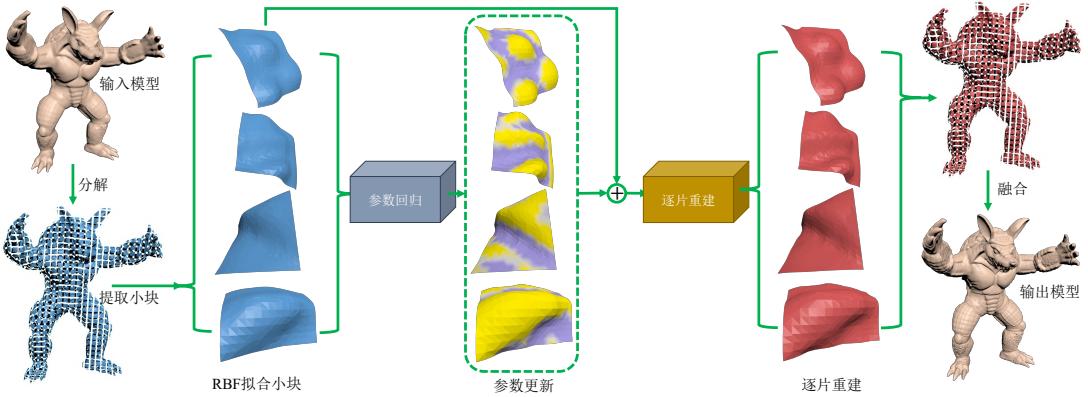


图 3.2 本文提出的表示及整体的算法流程。输入模型被分解为具有重叠的曲面小块。每个曲面小块都可以转化为 RBF 表示，并传入由两个子网络组成的 GDR-Net。从输入的 RBF 参数，第一个子网络预测参数空间中的位移，然后将其加到输入 RBF 参数中以形成细节恢复后的 RBF 参数。图中的 ‘+’ 表示参数空间中的加法运算。然后第二个子网络通过预测 DMCL 表示的占用概率和体素偏移来重建每个小块。最后，本文将所有重建的网格小块进行融合，从而形成一个完整的结果模型，即为恢复了细节的模型。

块的并集所覆盖。注意，在下面的描述中，本文总是用  $n$  来表示整个体数据的分辨率，用  $k$  来表示块的大小。

为了降低训练难度，本文将所有选中块的坐标通过平移转换为  $[0, k]^3$ 。当用 RBF 拟合每个小块时，每个块中的  $(k+1)^3$  个单元格节点自然会被选择为基函数  $\{\mathbf{c}_i\}$  的中心  $i = 1, 2, \dots, (k+1)^3$ 。在求解 RBF 参数时，本文使用  $(k+1)^3$  个单元格节点以及网格和单元边缘之间的交点作为样本点。因此，本文将每个小块  $f_{in}(\mathbf{x})$  拟合为以下形式：

$$f_{in}(\mathbf{x}) \approx \sum_{i=1}^{(k+1)^3} \bar{\lambda}_i \phi(|\mathbf{x} - \mathbf{c}_i|), \forall \mathbf{x} \in [0, k]^3. \quad (3.3)$$

这里， $\{\bar{\lambda}_i\}$  是拟合参数。本文取  $\phi(r)$  为：

$$\phi(r) = \frac{1}{\sqrt{2\pi}} \exp^{\frac{-r^2}{2\sigma^2}}, \quad (3.4)$$

并在所有实验期间将  $\sigma$  设置为 1.0。

## (2) 逐片细节恢复

GDR-Net 由两个子网络组成：一个用于参数回归，另一个用于重建。以  $\{\bar{\lambda}_i\}$  作为输入，第一个子网回归 RBF 参数空间中的参数更新  $\{\delta\bar{\lambda}_i\}$ 。然后，本文可以将修复后的形状表示为：

$$f_{out}(\mathbf{x}) = \sum_{i=1}^{(k+1)^3} (\bar{\lambda}_i + \delta\bar{\lambda}_i) \phi(|\mathbf{x} - \mathbf{c}_i|), \forall \mathbf{x} \in [0, k]^3. \quad (3.5)$$

在训练时，本文可以使用真实值  $f_{gt}$  作为监督。以前的工作<sup>[66,128]</sup>从隐式函数  $f_{out}(\mathbf{x})$  中提取重建网格，其中包含不可微的 Marching Cubes 算法<sup>[37]</sup>。为了使

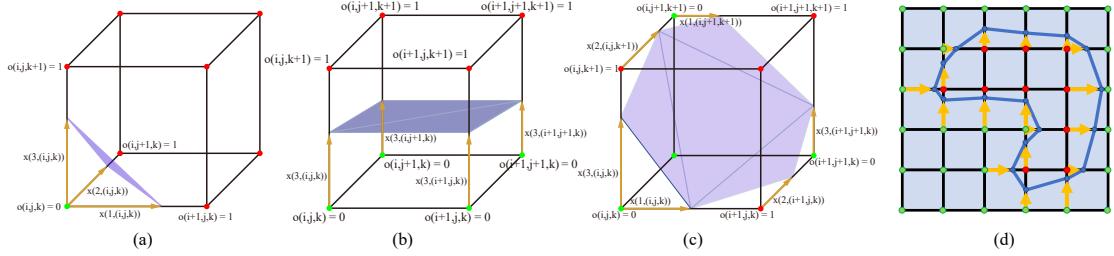


图 3.3 可微行进立方体场 (Deep Marching Cubes Layers, DMCL)。本文在 (a)、(b) 和 (c) 中展示了三种类型的单元拓扑。DMCL 由单元格节点上定义的占用概率 (记为  $o$ ) 和体素偏移量 (3 维, 记为  $x$ ) 组成。占用概率为 0 的绿色单元格节点位于网格外, 占用概率为 1 的红色单元格节点位于网格内部。单元格节点  $(i, j, k)$  处的三个偏移值表示单元格节点沿三个边缘方向到网格曲面的距离。对于不与网格曲面相交的边, 它们的偏移值设置为 0.5。(d) 展示了从 2D 视角中的 DMCL 表示重建网格的一个示例。占用概率 (绿色和红色单元格节点) 决定拓扑, 体素偏移值 (黄色箭头) 决定几何。

框架端到端可训练, 本文使用第二个子网络直接重建输出网格小块。本文使用 DMCL<sup>[17]</sup> 作为解码器的最后一层来表示输出的网格小块。DMCL 由占用概率 (记为  $\mathbf{O}_{out}$ ) 和体素偏移量 (记为  $\mathbf{X}_{out}$ ) 组成, 以显式和可微分的形式表示重建的网格形式。图3.3 中给出了更多关于 DMCL 的介绍信息。关于网络结构和损失项设计的细节将在 3.2.3 中讨论。

### (3) 网格融合

经过细节逐片恢复过程, 本文为每个小块得到了输出  $\mathbf{O}_{out}$  (形状为:  $((k + 1), (k + 1), (k + 1))$ ) 和  $\mathbf{X}_{out}$  (形状为:  $(3, (k + 1), (k + 1), (k + 1))$ )。在最后一步中, 本文将所有网格块融合在一起以获得整个占用概率场  $\mathbf{O}_{out}$  (形状为:  $(n + 1, n + 1, n + 1)$ ) 和整个体素偏移量  $\mathbf{X}_{out}$  (形状为:  $(3, n + 1, n + 1, n + 1)$ ) 的整个物体。

**占用概率融合。**对于每个单元格节点  $\mathbf{p}$ , 本文首先通过对来自不同块的占用概率值进行平均来计算其占用值  $\bar{\mathbf{o}}_{out}(\mathbf{p})$ 。然后, 本文通过解决以下优化问题来对占用概率值进行平滑:

$$\mathbf{O}_{out}^* = \arg \min_{\mathbf{O}} E_{occ}(\mathbf{O}), \quad (3.6)$$

其中

$$E_{occ}(\mathbf{O}) = \sum_{\mathbf{p} \in \mathbf{C}} |\mathbf{o}(\mathbf{p}) - \bar{\mathbf{o}}_{out}(\mathbf{p})| + w_{occ} \sum_{\|\mathbf{p}-\mathbf{q}\|=1} (\mathbf{o}(\mathbf{p}) - \mathbf{o}(\mathbf{q}))^2. \quad (3.7)$$

第一项鼓励输出占用概率接近预先计算的平均占用概率值。需要注意的是,  $\mathbf{C}$  表示至少被一个块覆盖的一组单元格节点。第二项鼓励相邻单元格节点具有接近的占用值。最后, 本文将  $\mathbf{O}_{out}^*$  值二值化以获得最终占用概率张量  $\mathbf{O}_{out}$ , 将小于 0.5 的值设置为 0, 其余的设置为 1。

**体素偏移量融合。**对于  $\mathbf{x}_{out}(d, \mathbf{p})$ , 其中  $d \in \{1, 2, 3\}$  和  $\mathbf{p} \in \{0, \dots, n\}^3$ , 本文使用  $\delta(d, \mathbf{p})$  来表示使用多少个值来计算  $\mathbf{x}_{out}(d, \mathbf{p})$ , 并将提供的值表示为  $\mathbf{x}_i(d, \mathbf{p}), i \in \{1, \dots, \delta(d, \mathbf{p})\}$ 。本文需要判断每个  $\mathbf{x}_i(d, \mathbf{p})$  的可靠性。每个体素偏移量值都定义

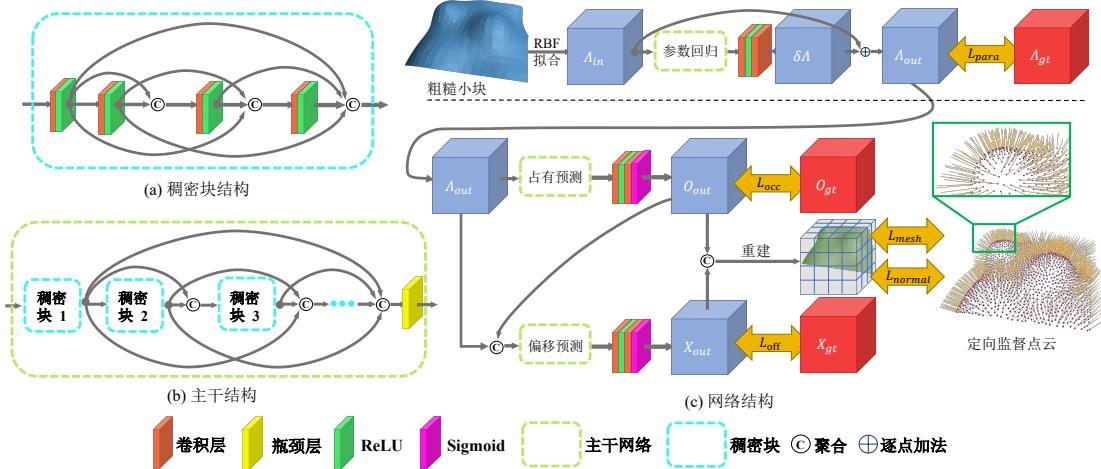


图 3.4 本文的 GDR-Net 的结构。(a) 说明了密集块的结构，其中所有层的特征都是密集连接的。(b) 说明了由几个密集块和一个瓶颈层组成的主干结构。不同密集块的输出特征也是密集连接的。整个网络结构如(c) 所示。用于参数回归的子网络展示在虚线上方，使用一个前向分支回归参数更新。虚线下方的解码器有两个分支，一个用于占用概率的回归，另一个用于体素偏移量的回归。在两个分支的最后都使用了一个 Sigmoid 层。损失函数项在图中展示为黄色双向箭头。

在具有起始单元格节点和结束单元格节点的边上。本文将  $\mathbf{x}_{out}(d, \mathbf{p})$  的开始和结束单元格节点上的占用概率表示为  $o_s(d, \mathbf{p})$  和  $o_e(d, \mathbf{p})$ 。 $\mathbf{x}_i(d, \mathbf{p})$  的开始和结束单元格节点上的占用概率记为  $o_{s_i}(d, \mathbf{p})$  和  $o_{e_i}(d, \mathbf{p})$ 。在最后一步，本文得到  $\mathbf{x}_{out}^*(d, \mathbf{p})$ :

$$\mathbf{x}_{out}^*(d, \mathbf{p}) = \sum_{i=1}^{\delta(d, \mathbf{p})} w_i \mathbf{x}_i(d, \mathbf{p}) / w. \quad (3.8)$$

这里， $w_i = (1 - |o_{s_i}(\mathbf{p}) - o_s(\mathbf{p})|) \cdot (1 - |o_{e_i}(\mathbf{p}) - o_e(\mathbf{p})|)$ ，并且  $w = \sum_i^{\delta(d, \mathbf{p})} w_i$ 。这些权重  $w_i$  可以衡量  $x_i(d, \mathbf{p})$  的可靠性。

一旦获得了整个物体的  $\mathbf{O}_{out}$  和  $\mathbf{X}_{out}$ ，就可以用它们来表示整个输出网格  $M_{out}$ 。结果展示在 3.3 中。

### 3.2.3 网络结构

在本节中，本文将描述网络结构和损失函数设计。如(2)所述，GDR-Net 由两个子网组成，分别记为  $g_1$  和  $g_2$ 。

#### (1) RBF 参数回归

$g_1$  的输入为  $\{\bar{\lambda}_i\}, i = 1, 2, \dots, (k+1)^3$ ，可以看作是一个三维张量  $\Lambda_{in}$ ，形状为  $((k+1), (k+1), (k+1))$ 。类似地，本文将参数更新  $\{\delta\lambda_i\}$  视为  $\delta\Lambda$  并将真实值参数记为  $\Lambda_{gt}$ 。 $g_1$  的公式可以写成：

$$\delta\Lambda = g_1(\theta_1, \Lambda_{in}), \quad (3.9)$$

其中  $\theta_1$  表示  $g_1$  中的权重参数。

对于网络结构，本文使用 DenseNet<sup>[133]</sup> 作为主干网络。DenseNet 最重要的组件是密集块。如图3.4(a) 所示，密集块中的层是密集连接的，有助于充分利用特征。同样，不同密集块的输出特征也是密集连接的，如图3.4(b) 所示。 $g_1$  的图示在图3.4(c) 的上半部分。在经过一些密集块和卷积层之后， $g_1$  输出  $\delta\Lambda$ 。

考虑到本文有真实值参数  $\Lambda_{gt}$ ，本文直接监督  $\delta\Lambda$ （带有正则化项）：

$$L_{regress} = w_1 L_{para} + w_2 L_{reg}, \quad (3.10)$$

其中  $w_i (i = 1, 2)$  是损失项的权重：

$$L_{para} = \|\Lambda_{in} + \delta\Lambda - \Lambda_{gt}\|^2, \quad (3.11)$$

$$L_{reg} = \|\delta\Lambda\|^2. \quad (3.12)$$

## (2) 逐片重建

第二个子网络  $g_2$  将修复的 RBF 参数作为输入，并将它们解码为输出占用概率和体素偏移量。公式可以写成：

$$\mathbf{O}_{out}, \mathbf{X}_{out} = g_2(\theta_2, \Lambda_{in} + \delta\Lambda), \quad (3.13)$$

其中  $\theta_2$  表示  $g_2$  中的权重参数。网络结构如图3.4(c) 的下半部分所示。 $g_2$  的主干与  $g_1$  中的相同；唯一的区别是  $g_2$  使用两个分支。一个分支用于预测占用率，另一个用于预测偏移量。预测的占用率被连接到偏移分支中，以帮助更好地预测偏移。两个分支都使用 Sigmoid 层来使输出值在  $[0, 1]$  范围内。

输出的  $\mathbf{O}_{out}$  和  $\mathbf{X}_{out}$  可以由它们的真实值  $\mathbf{O}_{gt}$  和  $\mathbf{X}_{gt}$  直接监督。此外，本文还在曲面上采样了一些定向点以进行监督。损失函数可以写成：

$$L_{reconstruct} = w_3 L_{occ} + w_4 L_{off} + w_5 L_{mesh} + w_6 L_{normal}, \quad (3.14)$$

其中  $w_i (i = 3, 4, 5, 6)$  是损失项的权重。损失函数的各项解释如下。

**占用概率损失项。**本文将  $L_{occ}$  作为  $\mathbf{O}_{out}$  和  $\mathbf{O}_{gt}$  之间的二元交叉熵损失：

$$L_{occ} = - \sum_{\mathbf{p}} (\mathbf{o}_{gt}(\mathbf{p}) \cdot \log(\mathbf{o}_{out}(\mathbf{p})) + (1 - \mathbf{o}_{gt}(\mathbf{p})) \cdot \log(1 - \mathbf{o}_{out}(\mathbf{p}))). \quad (3.15)$$

该损失项有助于网络预测输出网格的基本拓扑结构。

**体素偏移量损失项。**虽然重建网格的拓扑可以基于  $L_{occ}$  恢复，但仍然需要确定其偏移值以使重建网格接近真实值。本文计算预测的偏移量和真实偏移量之间的  $\ell_1$  损失：

$$L_{off} = \|\mathbf{1}_{gt} \cdot (\mathbf{X}_{out} - \mathbf{X}_{gt})\|_1, \quad (3.16)$$

其中  $\mathbf{1}_{gt}$  表示二进制掩码。在  $\mathbf{X}_{gt}$  中，一些偏移值实际上并不代表曲面上的一个点，因为一些边不与网格曲面相交。本文使用  $\mathbf{1}_{gt}$  来指定点是否在曲面上，并使用  $\cdot$  来表示逐点乘法。换句话说，实际上并不代表曲面上一个点的偏移值将不会被计入损失函数中。

**从真实值到重建网格的距离。**如图3.4所示，在每个单元格中，本文在真实值网格上密集采样点以用于监督。第  $\mathbf{r}$  个单元的采样点集记为  $Z_r$ 。本文将所有单元格的点到网格损失计算为：

$$L_{mesh} = \sum_{\mathbf{r}} \sum_{\mathbf{z} \in Z_r} d(\mathbf{z}, M(\mathbf{r}, \mathbf{X}_{out})), \quad (3.17)$$

其中  $M(\mathbf{r}, \mathbf{X}_{out})$  是由第  $\mathbf{r}$  个单元格中的偏移  $\mathbf{X}_{out}$  决定的曲面小块。本文使用  $d(\cdot, \cdot)$  来表示点到网格的距离， $L_{mesh}$  旨在最小化从真实值到重建网格的距离。因此， $L_{off}$  和  $L_{mesh}$  鼓励重建的网格和真实值网格彼此靠近。

**法向约束。**每个监督点都有一个定向法向，该法向应接近重建网格上其最近面的法向。法向的损失定义为：

$$L_{normal} = \sum_{\mathbf{r}} \sum_{\mathbf{z} \in Z_r} \|\psi(\mathbf{z}), \psi(M(\mathbf{r}, \mathbf{X}_{out}))\|^2. \quad (3.18)$$

这里， $M(\mathbf{r}, \mathbf{X}_{out})$  是由第  $\mathbf{r}$  个单元格中的偏移量  $\mathbf{X}_{out}$  决定的面片。此外， $\psi(\mathbf{z})$  表示点  $\mathbf{z}$  的法向，而  $\psi(M(\mathbf{r}, \mathbf{X}_{out}))$  表示在  $M(\mathbf{r}, \mathbf{X}_{out})$  中最接近点  $\mathbf{z}$  的面片的法向。 $L_{normal}$  的目的是鼓励重建网格的法向接近真实值网格的法向。

### 3.3 实验结果和讨论

本节介绍了实现细节、实验结果和相关分析、消融实验、与现有方法的比较以及鲁棒性测试。

#### 3.3.1 实现细节

**数据集。**本文的数据集包括合成模型和真实的扫描模型。对于合成数据，本文收集了 18 个高质量、细节丰富的网格，其顶点约为  $10^6$ - $10^7$ 。对于真实的数据，本文收集了 5 个细节丰富的物体，高度为 31-63 厘米。然后本文为本文的数据集生成了相应的高质量和粗扫描的网格对。为了测试泛化能力，本文在未见过的真实扫描数据集上测试了本文的方法，包括包含一般物体真实扫描模型的 Redwood<sup>[116]</sup> 数据集和人体真实扫描数据集 BUFF<sup>[115]</sup>。本文还对低成本扫描设备获得的一些极低质量的输入进行了测试。

本文使用扫描精度可调的 *EinScan-pro+* 扫描仪扫描本文收集的真实物体。在最精确模式下，扫描点之间的平均最小距离约为 0.7mm。本文使用最精确的扫



图 3.5 数据集中的三维模型示例。(a) 中列出了训练模型, (b) 中列出了测试模型。真实扫描模型以蓝色渲染, 而合成扫描模型以棕色渲染。

描结果作为真实值。然后本文将精度改为  $2mm$ , 得到相应的低质量粗糙扫描结果。

对于合成扫描模型, 本文将原始网格视为高质量网格, 并使用虚拟扫描仪对其进行扫描以生成粗糙扫描结果。具体来说, 本文使用两种类型的虚拟扫描仪来生成点云。首先, 本文使用基于射线的扫描方法, 在模型周围设置了 218 的视角方向。在每个方向上, 平行光线与模型曲面相交以获得扫描点。在组合不同视角的点后, 本文在点云的法向方向上添加了一些高斯噪声。本文还采用了基于散斑模拟和深度恢复的方法来获取点云。本文在模型周围设置了 218 的视点, 在每个视点生成模拟散斑图像, 然后从散斑图像中恢复深度图。对于这两种合成扫描方法, 扫描点均作为后处理进行统一下采样, 这有助于控制合成扫描精度。

本文使用 12 个模型 (10 个合成扫描模型和 2 个真实扫描模型) 作为训练集, 11 个模型 (8 个合成扫描模型和 3 个真实扫描模型) 作为测试集。对于数据增强, 本文将实际扫描的网格对旋转到 4 个不同的位置, 将合成模型在不同的位置扫描 2 次, 使用不同的扫描策略和随机扫描参数, 包括平行射线的密度、高斯的尺度噪声, 以及均匀采样的网格大小。

对于 12 个训练模型, 本文构建了 28 个网格对并提取了  $5.6 \times 10^4$  的小块对

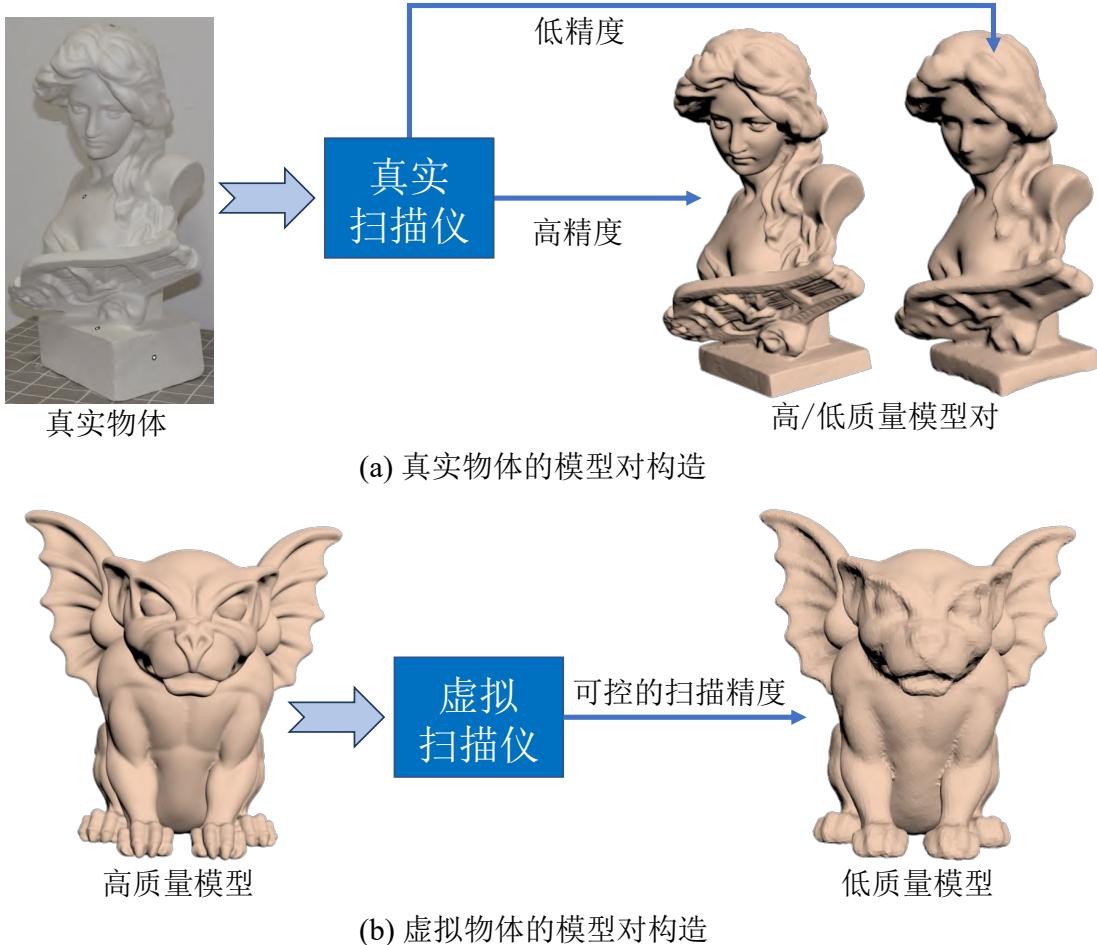


图 3.6 构建高/低质量网格对的过程。在 (a) 中，一个真实的物体被一个具有不同扫描精度的扫描仪扫描，以构建高质量/低质量的网格对。在 (b) 中，本文将具有丰富细节的原始网格视为高质量模型，并使用虚拟扫描仪对其进行扫描以实现低质量模型。

进行训练。由于本文的小块策略，所需的训练网格数量显著减少。训练和测试模型的例子如图3.5。构建高/低质量网格对的过程如图3.6所示。

**实验设置。**在本文的实验中，本文将数据集的分解分辨率  $n$  设置为 512，将块大小  $k$  设置为 16。结果网格的重建分辨率也是 512。对于网络结构，本文在主干中使用四个密集块。增长率为 16，这意味着一个密集块中第  $i$  个卷积层的输出通道数为  $16 \times i$ 。此外，本文在每个密集块中使用四个卷积层。在 (1) 和 (2) 中的损失函数包括六个权重  $w_i, (i \in \{1, 2, 3, 4, 5, 6\})$ ，它们分别设置为 1.0、0.1、10.0、5.0、5.0 和 3.0。在训练阶段，批大小设置为 25。学习率最初设置为  $10^{-5}$ ，在每 4000 次迭代后将其除以 2.0。本文使用 20000 次迭代来训练参数回归子网络，并进一步使用 20000 次迭代来一起训练两个子网络。所有训练和测试均在配备 Intel(R) Xeon(R) CPU E5-2687W v4 @ 3.00GHz、256GB RAM 和 8 个 12G TITAN Xp GPU 的工作站上运行。

**计算时间。**下面，本文给出每一步的计算时间。考虑到每个模型的结构和分辨率不同，以 *Owl* 模型（图3.5(b) 中的第一个模型）为例来展示计算时间。给定分辨

率为  $512^3$  的输入网格和 SDF 数据，提取 1698 个重叠小块需要 22 秒，使用 RBF 参数拟合小块需要 39 秒。逐个小块的优化和重建需要 43 秒，而融合步骤需要 39 秒。

**评估指标。**为了定量评估重建网格的质量，本文首先给出评估指标。令  $M_c$  表示要比较的网格， $M_{gt}$  表示真实值网格。对于  $M_{gt}$  和  $M_c$ ，本文在它们的曲面上采样点集  $S_{gt}$  和  $S_c$ ， $|S_{gt}| = |S_c| = 10^7$ 。使用这些样本点，本文可以计算倒角距离 (Chamfer Distance, CD) 和豪斯多夫距离 (Hausdorff Distance, HD)。对于  $M_c$  的高阶评估指标，本文评估  $M_{gt}$  和  $M_c$  之间的平均法向夹角 (MNA):

$$MNA(M_{gt}, M_c) = \frac{1}{|S_{gt}|} \sum_{v \in S_{gt}} \Delta_n(v, M_c) + \frac{1}{|S_c|} \sum_{v \in S_c} \Delta_n(v, M_{gt}), \quad (3.19)$$

其中  $\Delta_n(,)$  表示该点与其在网格上最近的面之间的法向角度。此外，本文还对评估模型的高曲率区域感兴趣。具体地，本文计算  $M_{gt}$  的每个顶点  $v$  的曲率水平，作为其两个主曲率的绝对值之和：

$$CL(v) = |k_1(v)| + |k_2(v)|, \forall v \in V(M_{gt}), \quad (3.20)$$

其中  $k_1$  和  $k_2$  是两个主曲率， $V(M_{gt})$  表示  $M_{gt}$  的顶点集。对于高曲率区域，本文通过计算点到网格距离来评估真实值顶点与比较重建网格之间的距离：

$$d_{CL \geq \alpha}(M_{gt}, M_c) = \frac{1}{|V_{gt, CL \geq \alpha}|} \sum_{v \in V_{gt, CL \geq \alpha}} d(v, M_c), \quad (3.21)$$

其中  $V_{gt, CL \geq \alpha}$  表示属于  $V(M_{gt})$  并满足  $CL(v) \geq \alpha$  的顶点  $v$  的集合。请注意， $d(,)$  表示一个点与最近的网格面之间的距离。综上，本文评估网格之间的距离和高阶法向角度，且兼顾了整个模型和局部高曲率区域。对于所有这些指标，越低越好。

### 3.3.2 结果

**定性分析。**下面的部分展示和分析测试集中模型的定性结果。在图3.7 中，本文从测试网格中选择了一些具有不同类型局部形状的部分。生成的网格自然、精致且看起来舒适，而输入的网格相对粗糙。除了输入和输出网格，本文还提供了其他可视化方法来呈现细节恢复效果。

首先，本文列出了输入和输出网格的误差图，展示了它们与真实值网格的距离。如图3.7的第二列和第四列所示，输出的误差明显小于输入的误差，这意味着输出模型更好地代表了潜在曲面（真实值模型）。本文还在图3.7 的第五列中展示了顶点偏移。顶点偏移展示了从输出到输入的有符号距离，反映了模型如何从输入到输出变化。黄色区域表示位移向外，紫色区域表示位移向内。从顶点偏

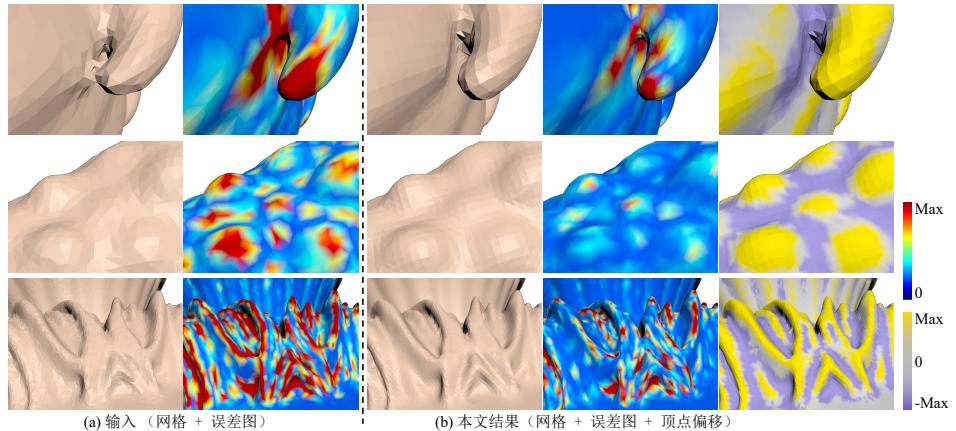


图 3.7 本文算法的结果。虚线左侧的模型是输入模型，而右侧的模型是本文的结果。本文还列出了网格的误差图，展示了网格与真实值模型之间的距离。对于本文生成的网格，本文还给出了顶点偏移，展示了从输出模型到输入模型的有符号距离。

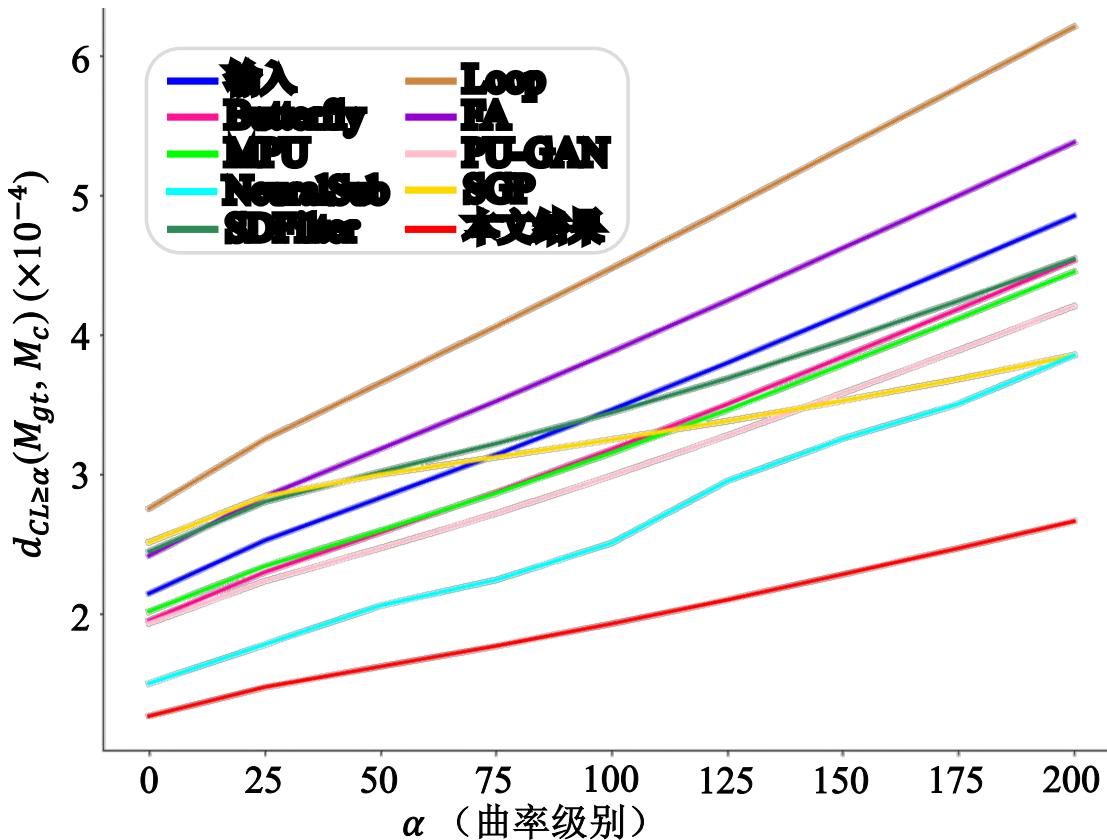


图 3.8 不同方法的定量比较。用不同颜色代表不同方法的结果，水平轴代表曲率水平，垂直轴代表高曲率区域的点到网格距离，随着曲率水平变大，本文结果明显优于其他结果。

移可以看出，凸的区域有向外扩张的趋势，凹的区域有向内收缩的趋势，平坦区没有明显的位移；这些结果共同表明本文的结果在保留低频特征的同时恢复了高频特征。

**定量分析。**定量结果在表3.1 中给出，其中最后一行展示了本文方法的结果。输出的 CD、HD 和 MNA 小于输入，这意味着输出模型比输入更能代表潜在曲面。此外，本文还在高曲率区域对它们进行了比较。在曲率水平  $\alpha$  的不同阈值下，

$d_{CL \geq \alpha}(M_{gt}, M_{out})$  总是小于  $d_{CL \geq \alpha}(M_{gt}, M_{in})$ 。更多分析见图3.8。这个问题的难度主要集中在高曲率区域，本文的输出在高曲率区域表现尤其好。

**表 3.1 不同方法的数量比较。比较的方法包括：Loop<sup>[121]</sup>, Butterfly<sup>[22]</sup>, FA<sup>[134]</sup>, Neural-Sub<sup>[135]</sup>, MPU<sup>[10]</sup>, PU-GAN<sup>[105]</sup>, SDFilter<sup>[26]</sup>, SGP<sup>[124]</sup>。所有的误差数值都是计算所有测试模型的平均值。**

度量 方法	CD ( $\times 10^{-7}$ ) ↓	HD ( $\times 10^{-4}$ ) ↓	MNA ( $\times 10^{-1}$ ) ↓	$d_{CL \geq \alpha} (\times 10^{-4}) \downarrow$								
				$\alpha=0$	$\alpha=25$	$\alpha=50$	$\alpha=75$	$\alpha=100$	$\alpha=125$	$\alpha=150$	$\alpha=175$	$\alpha=200$
Input	2.772	2.281	1.354	2.146	2.526	2.831	3.140	3.463	3.798	4.148	4.497	4.853
Loop	3.307	3.033	1.291	2.755	3.251	3.656	4.060	4.477	4.903	5.340	5.773	6.209
Butterfly	2.661	2.000	1.230	1.952	2.298	2.582	2.871	3.179	3.502	3.842	4.184	4.534
FA	3.002	2.666	1.938	2.417	2.841	3.182	3.524	3.879	4.245	4.623	4.998	5.379
NeuralSub	2.232	1.982	1.203	1.501	1.780	2.059	2.243	2.507	2.951	3.255	3.505	3.853
MPU	2.438	2.386	1.354	2.017	2.341	2.597	2.864	3.153	3.461	3.787	4.116	4.453
PU-GAN	2.274	2.293	1.291	1.932	2.234	2.470	2.718	2.989	3.277	3.582	3.889	4.205
SDFilter	3.517	3.434	1.356	2.447	2.803	3.018	3.220	3.442	3.687	3.957	4.243	4.545
SGP	3.397	4.215	1.354	2.514	2.842	2.998	3.124	3.249	3.382	3.527	3.683	3.854
本文 (DMCL)	2.454	2.026	1.300	1.656	1.908	2.081	2.256	2.447	2.653	2.876	3.103	3.337
本文 (RBF)	2.171	1.819	1.302	1.593	1.859	2.049	2.228	2.410	2.596	2.788	2.979	3.174
本文 (损失-1)	2.131	1.520	.260	1.337	1.547	1.695	1.861	2.027	2.208	2.399	2.618	2.861
本文 (损失-2)	2.085	1.430	1.285	1.281	1.492	1.646	1.805	1.963	2.161	2.369	2.568	2.791
本文	<b>2.046</b>	<b>1.350</b>	<b>1.160</b>	<b>1.267</b>	<b>1.474</b>	<b>1.622</b>	<b>1.769</b>	<b>1.929</b>	<b>2.100</b>	<b>2.283</b>	<b>2.470</b>	<b>2.663</b>

### 3.3.3 消融实验

为了分析每个组件的重要性，本文进行了消融实验。

**RBF 表示。**接下来探讨是否有必要使用 RBF 来拟合输入曲面小块。为此，本文删除了所有与 RBF 相关的过程并展示结果。正如本文对输出所做的那样，该消融实验尝试使用使用 DMCL 来表示输入曲面小块。所有其他实验设置与本文的主要实验相同。定量结果在表3.1 的第 10 行中给出。定性结果在图3.9 (a) 中给出。可以看到，定量效果不如以 RBF 参数为输入的策略。此外，在生成的网格上存在一些不平滑的区域，在图3.9 (a) 中用红色虚线椭圆标记。可以得出结论，从粗糙的 DMCL 回归修复的 DMCL 比回归 RBF 参数更新更困难。

**重建解码器。**本文还尝试删除第二个子网络并直接从修复的 RBF 系数中计算 DMCL 值。具体来说，对于每个小块，本文使用 RBF 计算有符号距离，并使用 Marching cubes 算法来获得占用和偏移量，然后使用牛顿法通过五次迭代来修复偏移值。注意到直接计算的偏移值不是 数值不稳定的（如<sup>[17]</sup>），本文只在这个消融实验中训练第一个子网络。结果如图3.9 (b) 和表3.1的第 11 行，性能不够好。这表明三维几何的直接监督比仅在 RBF 参数空间中的监督表现更好。

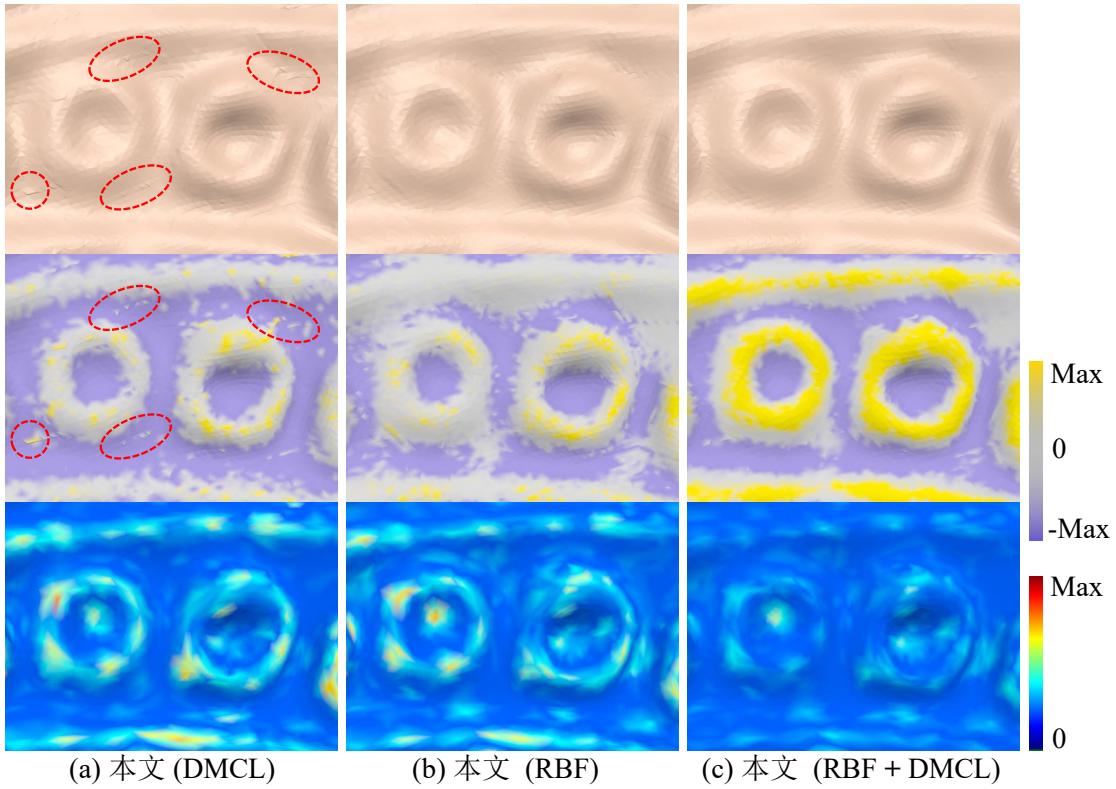


图 3.9 消融实验。第一行展示重建的网格。第二行和第三行展示了相应的顶点偏移和误差图。**(a)**展示了输入和输出均为 DMCL 的策略的结果。**(b)**展示了移除重建解码器的策略的结果。**(c)**部分展示了本文的框架的结果，该框架对输入的 RBF 参数进行更新，然后解码为输出 DMCL。

损失项。本文还测试了重建解码器的损失项的不同选择。本文尝试只使用  $L_{occ}$  和  $L_{off}$ ，将结果记录在表3.1 的第 12 行中（表示为“本文(损失-1)”）。然后本文又尝试使用  $L_{occ}$ 、 $L_{off}$  和  $L_{mesh}$ ，记录在表3.1 的第 13 行（表示为“本文(损失-2)”）。当使用公式(3.14) 中的所有项时，本文的框架实现了最佳性能。

**体素化分辨率。**本文还尝试了体素化分辨率  $n$  的不同选择。以 *Armadillo* 模型为例，本文应用实验设置  $n = 200, 400, 600, 800, 1000$ ，发现 CD 误差 ( $\times 10^{-7}$ ) 为 27.19, 7.80, **7.72**、7.76 和 7.93。当  $n$  太小时，体数据不具备很好地表示曲面的能力。当分辨率太大时，每个块的感受野太小，无法很好地预测丢失的细节。总之，本文算法可以调整  $n$  的值以确保获得满意的结果。本文为整个测试集设置了  $n = 512$ 。

### 3.3.4 比较

本文将本文方法与细分、点云上采样和细节增强方法进行比较。请注意，为了公平比较，本文确保细分和点云上采样结果具有比本文更多的顶点，所有基于学习的对比方法均和本文使用相同的训练集。细节增强的结果受任务本身约東，与输入的顶点数相同。定性比较结果如图3.10所示。定量比较在表3.1 中，高

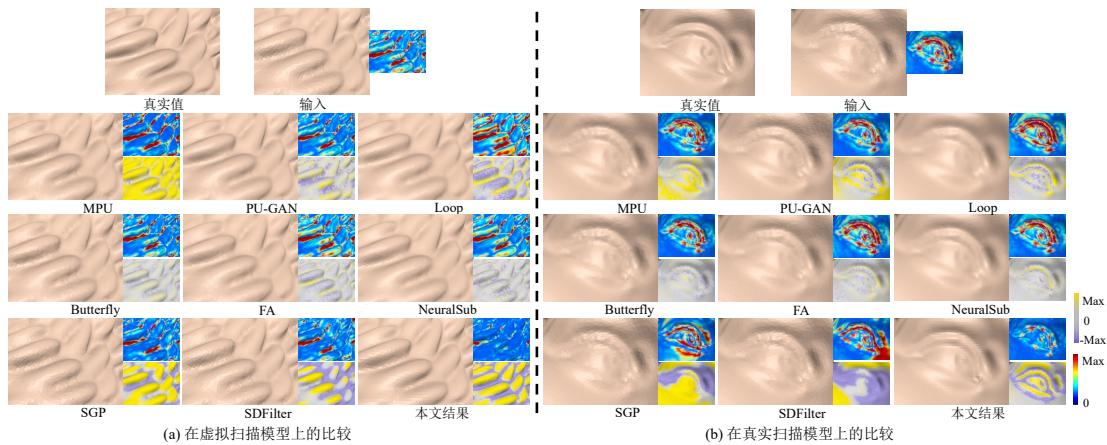


图 3.10 不同方法的定性比较。在 (a) 和 (b) 中, 上图展示了不同类型模型的一些局部视角 (左侧为合成扫描, 右侧为真实扫描)。给出了比较的网格、误差图和顶点偏移, 以便进行清晰的定性比较。

曲率区域更直观的比较在图3.8 中。

与细分方法的比较。本文将本文方法与四种细分方法进行比较，包括 Butterfly<sup>[22]</sup>、Loop<sup>[121]</sup>、特征自适应（Feature Adaptive, FA）<sup>[134]</sup>方法和神经细分（Neural Sub-division, 简称 NeuralSub）<sup>[135]</sup>。所有的细分结果都比本文的结果有更多的顶点。

在表3.1 和图3.10 中的误差图中，可以看到，四种方法都没有像本文的方法那样显著减少误差。这些方法的位移分布也不明显。只有 Butterfly 和 NeuralSub 的结果与本文的位移分布相似（在凸区域扩展，在凹区域收缩），而其他的趋势完全相反。

与点云上采样的比较。本文还将本文方法与在本文数据集上重新训练的点云上采样方法 MPU<sup>[10]</sup> 和 PU-GAN<sup>[105]</sup> 进行了比较。本文直接使用扫描点（用于重建输入网格）作为两种方法的输入，比例因子设置为 4。每个点的法向由 PCL 库估计<sup>[136]</sup> 然后向外定向。至于后处理，本文通过泊松重建方法从上采样点云重建结果网格<sup>[16]</sup>，深度为 12。

定性对比如图3.10。虽然顶点偏移与本文结果不冲突，但误差图并没有显著改善。而且，通过观察表3.1的第六行和第七行，可以发现其定量结果不如本文方法。

与细节增强的比较。本文还将本文方法与细节增强方法 SDFilter<sup>[26]</sup> 和 SGP<sup>[124]</sup> 进行了比较；这些方法首先对网格使用滤波，然后调整组合参数以获得令人满意的结果。SDFilter 和 SGP 都依赖于手动设置的参数，可能会产生不自然的结果。在实验中，本文调整其方法的参数以使结果看起来尽可能合理。为此，本文尝试了几组参数并选择其中最好的进行比较。

定性对比如图3.10。虽然本文为其微调了参数，但误差图仍然没有明显改善，如果不微调参数，结果会比输入差很多。从顶点偏移可以发现，位移分布不像本文的那么规则。在表3.1中，可以看到，结果在数量也上不如本文。请注意，结

果仅在高曲率区域比输入要好，如图3.8所示。虽然细节增强过程可以增强高曲率区域的细节，但不能很好地保持低曲率区域的形状。

### 3.3.5 鲁棒性和泛化性

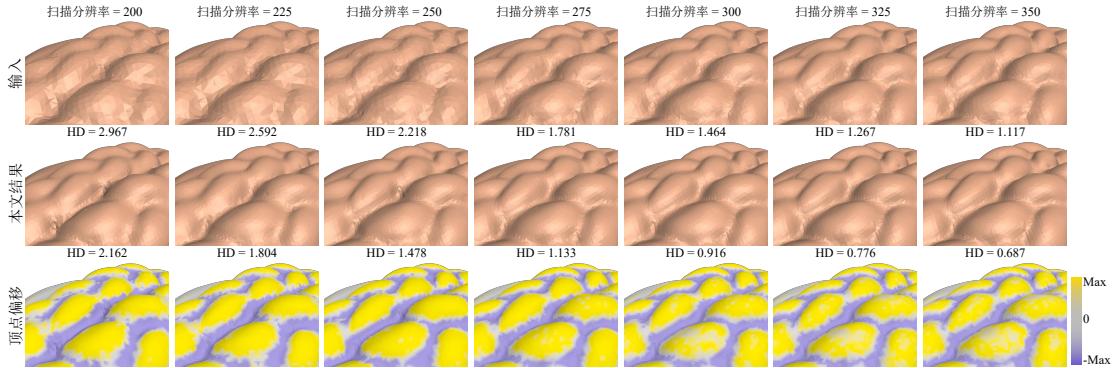


图3.11 对扫描分辨率的鲁棒性。第一行展示了使用不同扫描分辨率扫描的输入模型。第二行展示了本文的结果。第三行展示顶点偏移。上图还列出了每个模型的 HD 距离 ( $\times 10^{-4}$ )。

**对扫描分辨率的鲁棒性。**在实际应用中，扫描仪的扫描分辨率在不同的应用场景中可能会受到限制。为了测试不同扫描分辨率下的鲁棒性，本文用不同的扫描分辨率扫描同一物体。为了便于比较，本文对所有输出模型采用相同的重建分辨率(512)。如图3.11，当扫描分辨率降低时，扫描模型丢失更多细节，输出模型也更差。然而，通过观察顶点偏移，可以发现位移分布保持一致，这意味着不同扫描分辨率的细节恢复趋势是一致的，从而展示出鲁棒性。

**对物体类别的鲁棒性。**在[65-66]中已经证明，局部连续隐式表示对于不同类别的物体具有良好的泛化能力。本文还在3.1中讨论了扫描细节误差的局部性。现在本文通过将本文的框架应用于没有见过的数据集来实验性地测试算法的泛化能力。

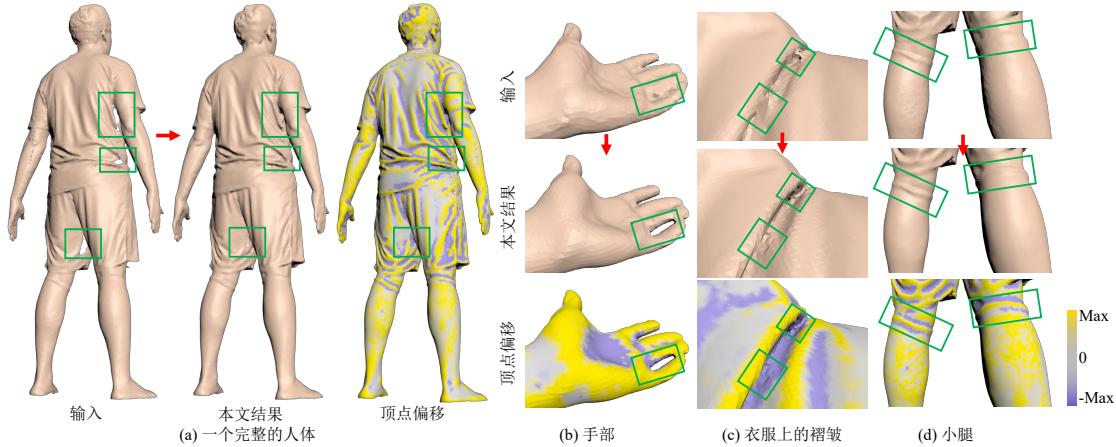


图3.12 真实扫描数据集 BUFF 上的细节恢复效果展示<sup>[115]</sup>。(a) 中的方框表明，即使输入模型有未闭合的孔，本文也可以获得自然且较好的结果。在(b) 和(c) 中，该部分展示了本文的框架允许更改拓扑。(d) 中的方框展示了在过滤小尺度局部噪声的同时，在特征丰富的区域中的细节增强效果。

图3.12展示了数据集 BUFF<sup>[115]</sup>上的测试结果，其中包含真实的扫描人体模型。本文将数据集中真实扫描模型作为输入，并使用本文的框架预测输出。在图3.12 (a) 中，可以看到与本文的测试模型中相似的位移分布，这表明本文的框架具有很强的泛化能力。即使输入有未闭合的孔，结果仍然是高质量的。本文还表明，如果需要，本文的框架允许拓扑更改。在图3.12 (b) 中，无名指和小指在输出中是分开的，而在输入中它们是粘在一起的。同样，在图3.12 (c) 中，衣服的褶皱中出现了意想不到的粘连，这些粘连在本文的输出中被移除。在图3.12 (d) 中，输入端的小腿上存在一些意想不到的小尺度噪声。本文的框架仍然可以自动增强特征丰富区域的细节（例如，在图3.12 (d) 中标记的框中的袜子和小腿的交界处）。

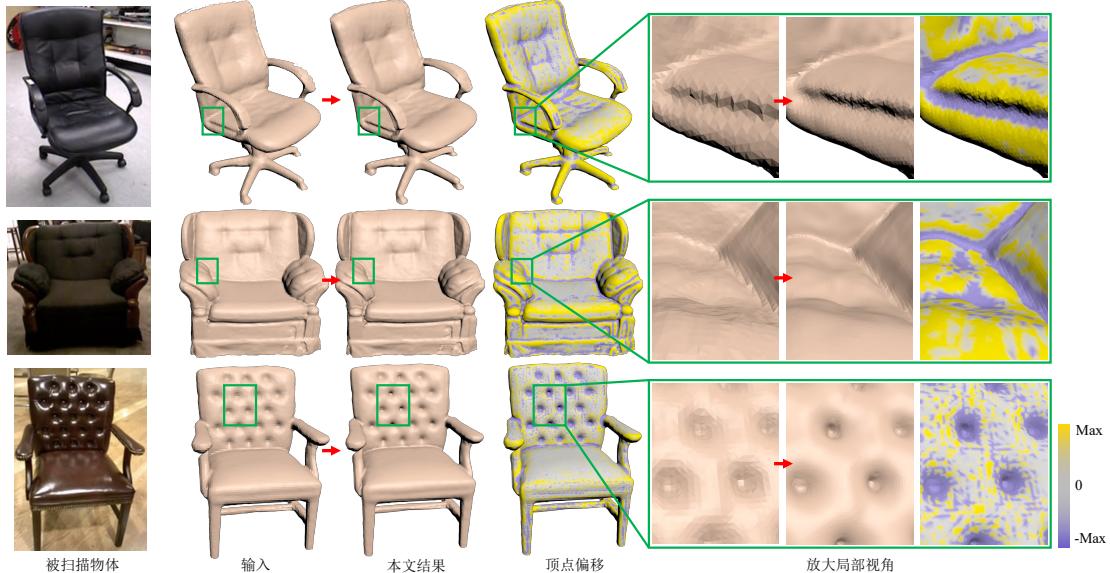


图 3.13 真实扫描数据集 Redwood<sup>[116]</sup>的上细节恢复效果展示。该图列出了原始扫描物体、输入模型、输出模型、顶点偏移和一些放大的局部部分。

本文还在数据集 Redwood<sup>[116]</sup>中测试了真实扫描的室内场景物体。结果如图3.13所示。整体位移分布趋势也与在本文自己的测试集上表现一致。此外，本文的方法过滤了小规模的局部噪声，使输出模型局部平滑且全局细节丰富。

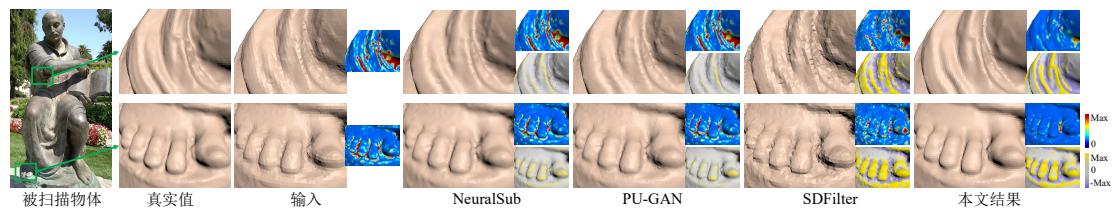


图 3.14 测试来自 Tanks and Temples<sup>[137]</sup> 数据集的 Ignatius 模型。第 3 到第 7 列的结果分别是 (8.6, 7.6, 2.4), (7.7, 7.3, 2.2), (7.8, 7.7, 2.2), (9.4, 8.9, 2.8) 和 (6.7, 7.1, 2.0)。

在不重新训练网络的情况下，本文还在真实的扫描数据集 Tanks and Temples<sup>[116]</sup>上进行了测试，该数据集包含真实值高质量模型和单视角扫描。本文融合下采样的单视角扫描以获得低质量的输入，结果如图3.14所示。为了进行比较，

本文为每个比较任务选择一种方法：NeuralSub（细分）、PU-GAN（点云上采样）和 SDFilter（细节增强）。对于 NeuralSub，本文将输入细分 1 次。PU-GAN 的上采样因子为 4。两种方法的结果都比本文的结果有更多的顶点，这确保了比较是公平的。本文的方法在定性和定量上都在没有见过的数据集上表现最好，证明了本文方法的泛化能力。

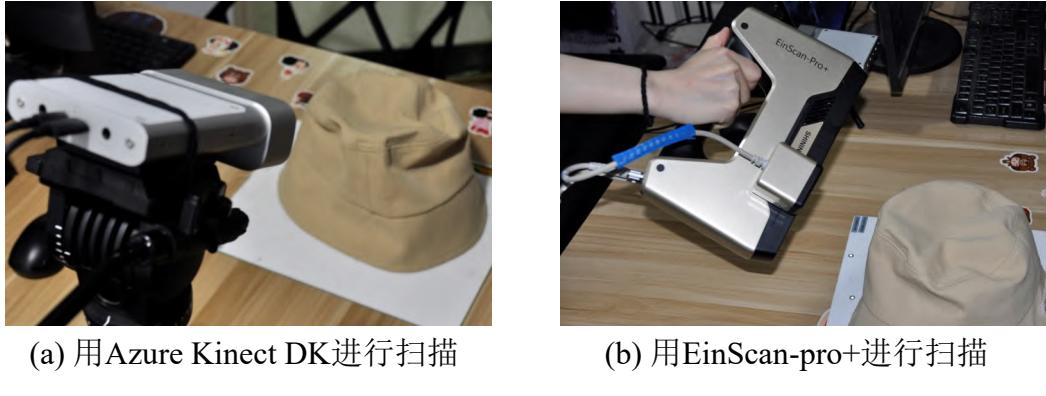


图 3.15 为了测试由低成本设备获得的更粗糙的输入，本文使用 *Azure Kinect DK*<sup>[7]</sup>（在 (a) 中，低质量）和 *EinScan-pro+*<sup>[8]</sup>（在 (b) 中，高质量）扫描相同的物体，作为高/低质量对。

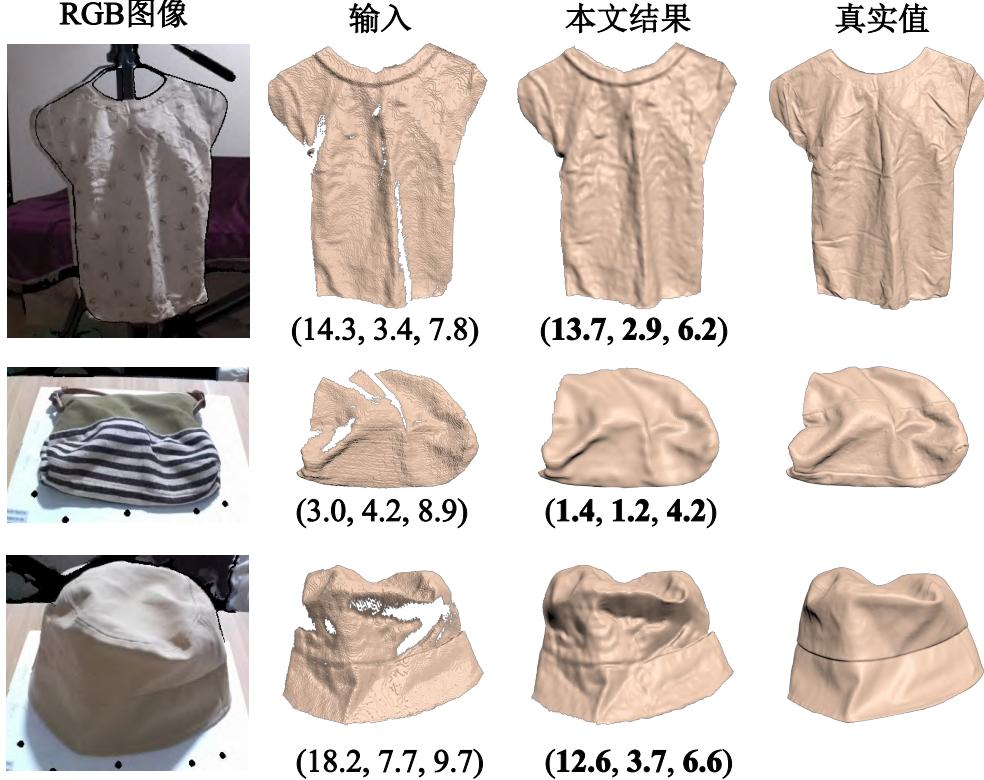


图 3.16 以低成本深度图作为输入的比较结果。评价指标包括 (CD( $\times 10^{-5}$ )、HD( $\times 10^{-2}$ )、MNA( $\times 10^{-1}$ ))。

**对低质量输入的鲁棒性。**为了进一步探索泛化能力，本文测试了低成本扫描设备获得的更低质量的输入模型。本文选择一些普通的物品而不是雕塑作品，包括衣

服、帽子和书包。如图3.15，本文通过 *EinScan-pro+*<sup>[8]</sup> 和 *Azure Kinect DK*<sup>[7]</sup> 分别得到高/低质量对。为了对齐它们，本文首先使用一些标记点进行粗对齐，然后应用 FRICP<sup>[12]</sup> 进行精确对齐。对于定量比较，不同方法使用相同的掩码裁剪输入、真实值和输出。结果及对比如图3.16，对比方法参数设置与其他实验相同。本文的方法在定性和定量上都表现最好，这证明了本文的方法在低成本深度数据上的有效性。

### 3.4 本章小结

本文提出了一种基于径向基函数的局部形状表示，适用于几何细节恢复任务。本文的网络称为 GDR-Net，用于恢复低质量扫描模型丢失的几何细节。该框架采用分治策略，利用扫描误差分布的局部性，降低了训练难度，使得处理不同大小的模型成为可能。此外，本文构建了合成和真实扫描模型的数据集，包括高/低质量对。大量实验证明了本文框架的有效性和稳健性。此外，本文通过顶点偏移分析了扫描误差的分布，为这项任务提供了其他思路。广泛的评估表明，本文的框架优于其他方法。

实验表明，基于本文所提出的表示的方法正确地恢复了几何细节并具有良好的泛化能力。此外，本文的框架对不同的扫描分辨率、看不见的物体类别，甚至低质量的输入（具有局部噪声和未闭合的孔）都具有鲁棒性。尽管取得了有希望的结果，但该方法仍有局限性。首先，本文的框架基于局部几何，因此无法整合语义特征。其次，如果噪声太大，本文的方法可能会将局部噪声误认为是局部特征。整合整体几何信息可能有助于网络处理这些限制。对每个物体类别单独进行训练并使用先验形状可以进一步缓解这些问题，但这种技术也会降低泛化能力。未来工作可以探索这些问题的潜在解决方案。除此以外，本文所提出的基于径向基函数的局部形状表示并不拘泥于三维模型的细节恢复，其同样适用于其他的任何需要高精度局部建模能力的三维几何任务，如高精度重建等。

## 第4章 基于逐点加权刚性形变的非刚性几何形变表示

### 4.1 引言



图 4.1 本文提出了一种新的非刚性形变，可以用在循环网络当中。通过使用循环单元，网络将源曲面进行迭代地形变，直到收敛到接近目标曲面。通过提出的多视角对齐二维投影损失函数，本文对齐形变曲面和目标曲面，并且整个过程是以无监督的方式进行训练的。

本章节研究曲面配准问题。曲面配准问题旨在寻找两个曲面之间的空间变换和对应关系，是计算机视觉和图形学中的一个基本问题，在三维重建<sup>[33-34]</sup>、跟踪<sup>[138-139]</sup>和医学成像<sup>[140-141]</sup>等任务中都有重要作用。正如在第一章的1.2.2中所介绍的，根据将源曲面形变到目标曲面的变换类型，配准算法可以分为刚性<sup>[67,77,80,82,84,142-143]</sup> 和非刚性<sup>[85,89-91,98]</sup>两种情形。刚性配准估计对齐两个曲面的全局旋转和平移，而非刚性配准具有更大的自由度，因此更加复杂和具有挑战性。

传统的基于优化的方法<sup>[67,85]</sup>通常通过迭代交替地寻找对应关系和进行形变对齐这两个步骤来处理这个问题。寻找对应关系的步骤是找到源曲面和目标曲面之间的对应点，而进行形变对齐的步骤则根据当前预测的对应关系来估计形变。然而，仅仅基于启发式（例如最近邻搜索）或人工设定的特征（如 SHOT<sup>[144]</sup> 和 FPFH<sup>[73]</sup>）构建可靠的对应关系并不容易。

最近，基于学习的方法通过利用神经网络强大的表示学习能力开始展示出可观的配准结果，但大多数都局限于刚性配准<sup>[77,80,82,84,142-143]</sup>。只有少数几种基于学习的非刚性配准方法<sup>[90,98]</sup>，通过预测逐点偏移场或者薄板样条 (Thin Plate Spline, TPS)<sup>[145]</sup>相关系数来合成非刚性形变。可以说，设计一个可以产生非常准确结果的基于学习的非刚性配准框架并不简单。

学习非刚性配准的挑战在于两个方面。首先，与单一的全局刚性变换不同，非刚性配准的更大自由度增加了网络训练的难度。例如，传统方法<sup>[85,146]</sup>估计曲面上每个点的局部变换。为了缓解这个问题，一个可能的方向是采用节点图<sup>[99]</sup>表示，它将形变的复杂度从所有曲面点级别降低到图节点数量级别。然而，由于形状相关的节点图的图节点数量和拓扑结构不同，将其传递给神经网络并不方

便。其次，带标签数据的缺乏限制了非刚性配准网络的训练。从真实的大规模数据中获得密集的非刚性对应关系以作为直接监督并非易事。另一种方法是以无监督的方式训练网络。然而，用于形状相似度测量的现有度量，例如倒角距离（Chamfer Distance, CD）和推土机距离（Earth Mover's Distance, EMD）等，不足以驱动网络学习正确的解决方案。

为了解决这些问题，本文首先提出了一种适用于网络学习的新的非刚性表示。具体来说，本文将非刚性变换表示为  $K$  个刚性变换的逐点组合，其中  $K$  远小于曲面点的数量。这是因为曲面的非刚性形变可以通过几个刚性变换的逐点组合很好地近似。这种表示不仅使本文的方法能够近似任意非刚性变换，而且使解空间受到良好约束。为了学习这种表示，本文设计了一个循环神经网络架构来迭代地估计组合权重和每个刚性变换。在每次迭代中，网络只需要估计一个单一的刚性变换和每个点的蒙皮权重，它代表了刚性变换对该点的影响程度。虽然在神经网络中已经采用了迭代策略进行配准<sup>[80-81]</sup>，它们主要用于刚性配准，而本文提出的迭代方法专为非刚性配准而设计，同时也兼容刚性配准。

本文进一步提出了一种多视角损失函数，以自监督的方式训练模型。具体来说，本文将三维曲面投影到多视角二维深度图像，并通过深度图像和掩码图像来衡量源曲面和形变曲面的视觉相似度。直观的想法是，如果两个曲面完全配准，它们的投影深度图和掩码图应该是相同的。此外，本文采用软光栅化方法将点云渲染为深度图和掩码图，这使得本文的损失项是可微的。本文的自监督学习方法采用了所提出的形状相似度损失，它优于常用的 CD 和 EMD 度量。

结合以上两点，本文将所提出的网络称为循环多视角对齐网络（Recurrent Multi-view Alignment Network, RMA-Net）。本文对不同的物体类型（身体、人脸、猫、狗和 ModelNet40 数据集<sup>[147]</sup>）进行了广泛的实验。本文提出的 RMA-Net 不仅优于以前最先进的方法，而且处理较大形变程度的非刚性配准也能很好地工作。广泛的消融实验证明了本文提出的方法中每个组件的有效性。

综上，这项工作的贡献包括：

- 本文提出一种基于逐点加权刚性形变的非刚性几何形变表示，且设计了循环多视角对齐网络，使得这个表示在神经网络中得以使用。
- 本文提出了一种自监督的多视角损失函数，将三维曲面投影到多视角二维图像，并在二维图像上进行相似度的度量计算。
- 实验证明本文提出的方法效果优于之前的方法。

## 4.2 基于逐点加权刚性形变的非刚性几何形变表示：设计与应用

### 4.2.1 问题描述

给定源点云  $\mathcal{S} \in \mathbb{R}^{M \times 3}$  和目标点云  $\mathcal{T} \in \mathbb{R}^{N \times 3}$ , 本文的目标是找到一个非刚性变换  $\phi : \mathbb{R}^{M \times 3} \rightarrow \mathbb{R}^{M \times 3}$ , 使得形变点云

$$\tilde{\mathcal{S}} = \phi(\mathcal{S}) \quad (4.1)$$

尽可能接近目标点云  $\mathcal{T}$ 。本文的目标是设计一个基于学习的框架, 以源点云  $\mathcal{S}$  和目标点云  $\mathcal{T}$  作为输入, 直接预测非刚性变换  $\phi$ 。

### 4.2.2 基于逐点加权刚性形变的非刚性几何形变表示

本文提出用一系列刚性变换  $\{\psi_r\}_{r=1}^K$  的逐点组合来表示非刚性变换  $\phi$ :

$$\phi(\mathcal{S}) = \sum_{r=1}^K \mathbf{w}_r \cdot \psi_r(\mathcal{S}), \quad (4.2)$$

其中  $\mathbf{w}_r \in \mathbb{R}^{M \times 1}$  是逐点蒙皮权重,  $\cdot$  表示逐点乘法,  $K$  是远小于点数的刚性变换的数量。

对于每个点, 本文约束分配给所有刚性变换的权重, 强制性要求它们满足总和等于 1 的条件:

$$\sum_{r=1}^K \mathbf{w}_r(i) = 1, \quad \forall i = 1, 2, \dots, M. \quad (4.3)$$

当  $K = 1$  时, 它退化为刚性变换。当  $K \geq 2$  时, 每个曲面点受到多个蒙皮权重不同的刚性变换影响, 在这种情况下它可以表示非刚性变换。当  $K$  变大时, 其表示能力逐渐增强。

与基于节点图的表示相比, 本文的方法具有以下优点:

- 与在图节点上定义局部变换的节点图不同, 本文提出的模型不需要为每个特定曲面构建节点图, 如方程(4.2)中的刚性变换  $\psi_r$  是在曲面上的所有点全局定义的。
- 与给定曲面和节点图的固定蒙皮权重不同, 公式(4.2) 中的蒙皮权重是学习出来的, 并且可以根据不同的源曲面和目标曲面进行自适应调整。

这样的表示不仅可以表达复杂的非刚性形变, 还可以通过去除权重并更改公式(4.2) 中的加法为乘法来轻松扩展到刚性配准:  $\phi(\mathcal{S}) = \psi_K \circ \psi_{K-1} \circ \dots \circ \psi_1(\mathcal{S})$ 。

### 4.2.3 循环更新框架

然而, 公式(4.2) 中的变量数 ( $\{\psi_r\}_{r=1}^K$  和  $\{\mathbf{w}_r\}_{r=1}^K$ ) 仍然很大, 包括  $M \times K$  的蒙皮权重和  $6 \times K$  的刚性变换。因此, 同时直接预测所有变量可能不容易。为

了解决这个问题，本文提出了一种循环更新策略，以逐步回归刚性变换和蒙皮权重。在第  $k$  个阶段，形变点云表示为：

$$\mathcal{S}^k = \sum_{r=1}^k \mathbf{w}_r^k \cdot \psi_r(\mathcal{S}), \quad (4.4)$$

其中第  $r$  个刚性变换在第  $k$  个阶段的逐点权重表示为  $\mathbf{w}_r^k$  并且本文始终保持约束  $\sum_{r=1}^k \mathbf{w}_r^k = \mathbf{1}$  被满足。

接下来的部分介绍如何循环获得逐渐形变的点云  $\{\mathcal{S}^k\}_{k=1}^K$ 。在第 1 个阶段，本文的网络预测单个刚性变换  $\psi_1$ ，随之可以得到一个变换后的点云  $\mathcal{S}^1 = \mathbf{w}_1^1 \cdot \psi_1(\mathcal{S})$ ，其中  $\mathbf{w}_1^1 \equiv \mathbf{1}$ 。在第  $k$  阶段，当  $k \geq 2$  时，网络回归  $\psi_k$  和  $\mathbf{w}_k^k$ 。为了满足公式(4.3)中的约束，本文可以缩放之前阶段的预测权重 ( $\{\mathbf{w}_r^k\}_{r=1}^{k-1}$ ) 的因子为  $1 - \mathbf{w}_k^k$ ，因此可以得到以下更新公式：

$$\mathbf{w}_r^k = (\mathbf{1} - \mathbf{w}_k^k) \cdot \mathbf{w}_r^{k-1}, \quad 1 \leq r \leq k-1. \quad (4.5)$$

根据更新公式可以验证公式(4.3)在各个阶段都满足。据此可以推导出形变点云在各个阶段的递推公式：

$$\mathcal{S}^k = (\mathbf{1} - \mathbf{w}_k^k) \cdot \mathcal{S}^{k-1} + \mathbf{w}_k^k \cdot \psi_k(\mathcal{S}). \quad (4.6)$$

基于上述公式，本文的网络只需要预测一个刚性变换和每个阶段的蒙皮权重，这大大降低了学习的难度。通过这种方式，本文迭代地更新形变点云，使得形变点云序列逐渐收敛到目标。

#### 4.2.4 网络结构和损失函数

本节将详细介绍本文提出的循环多视角对齐网络（Recurrent Multi-view Alignment Network, RMA-Net）和损失函数。

##### (1) 网络架构

本文采用循环网络迭代更新形变点云。该网络包括两个关键组件。首先，提取输入源点云和目标点云的深层特征，并通过点积运算计算它们的相关度。其次，使用循环更新模块来实现迭代过程。本文在图4.2 中展示了完整网络结构。

**特征提取和相关计算。**类似于<sup>[82]</sup>，本文首先使用 DGCNN<sup>[19]</sup> 和 Transformer<sup>[20]</sup>从输入点云  $\mathcal{S}^{k-1}$  和  $\mathcal{T}$  中提取深度特征。特征的大小为  $M \times C$  和  $N \times C$ ，其中  $C$  是特征通道。然后使用这两个特征中每个特征向量之间的点积运算计算相关张量。相关度大小为  $M \times N$ 。接下来对相关度的最后一个维度执行最大 K 维提取操作，这消除了对目标点数  $N$  的依赖。生成的相关度与源特征具有相同的维度。在更

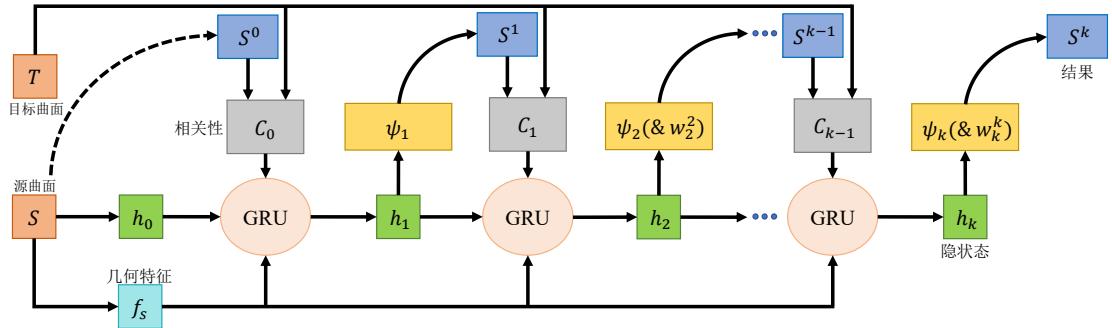


图 4.2 本文的完整网络结构示意图。在基于 GRU 的框架中，隐藏状态通过从源点云中提取特征进行初始化（表示为  $h_0$ ）并在后续阶段中反复更新。在第  $k$  阶段，本文提取当前形变点云  $S^{k-1}$  和目标点云  $T$  的相关度  $C_{k-1}$ 。源点云的几何特征  $f_s$  被提取出来，并与相关度聚合在一起作为每个阶段的更新单元的输入。最后本文从  $h_k$  回归一个刚性变换  $\psi_k$  及其对应的逐点蒙皮权重  $w_k^k$ 。

新期间使用源特征、全局目标特征（目标特征的平均池化和扩展）和相关度的聚合，表示为  $C_k$ 。

计算相关度的具体过程如图4.3所示。在使用 DGCNN<sup>[19]</sup> 和 Transformer<sup>[20]</sup> 提取特征之后（其中 DGCNN 有 5 个 EdgeConv 层，Transformer 有 4 个多头注意力中的头，特征通道号设置为 1024），本文对计算出的相关度的最后一个维度执行最大 K 维提取（Top-K）操作，以使输出与点数无关，其中 K 设置为 1024。然后本文将最大 K 维提取结果和两个点云的深度特征聚合在一起作为下一次迭代更新的输入。

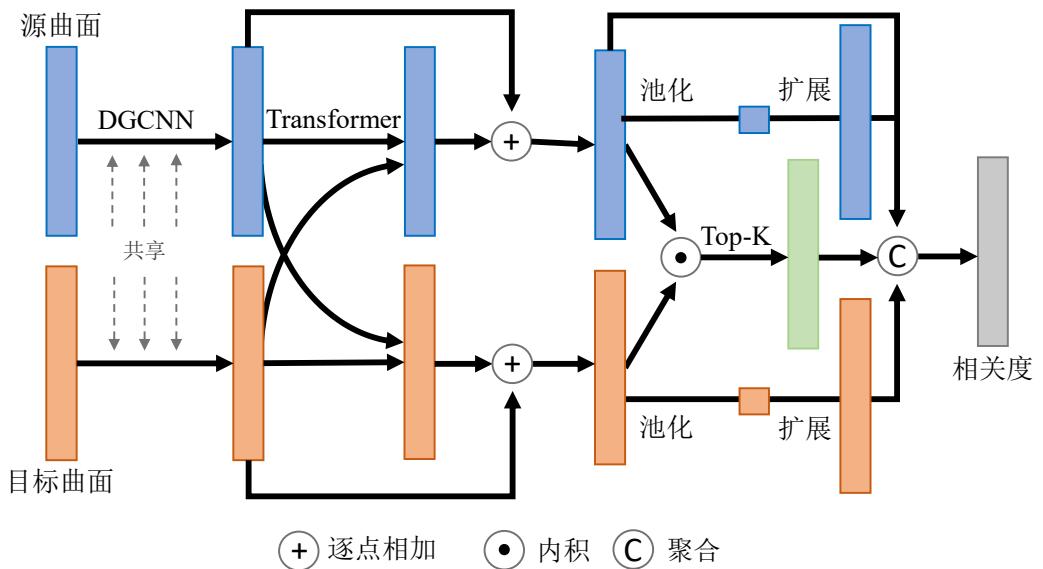


图 4.3 从形变点云和目标点云中提取相关度的过程。

**基于 GRU 的更新。**本文采用 GRU<sup>[148]</sup> 来实现4.2.3 中提到的循环更新过程。整体架构类似于最近的光流估计工作 RAFT<sup>[149]</sup>，但有几个重要的区别。首先，本文关注不规则点云而不是规则的像素点，因此完全相同的架构在这里不适用。其次，本文的关键贡献是一种新的非刚性表示，而循环框架仅仅是一个使本文的

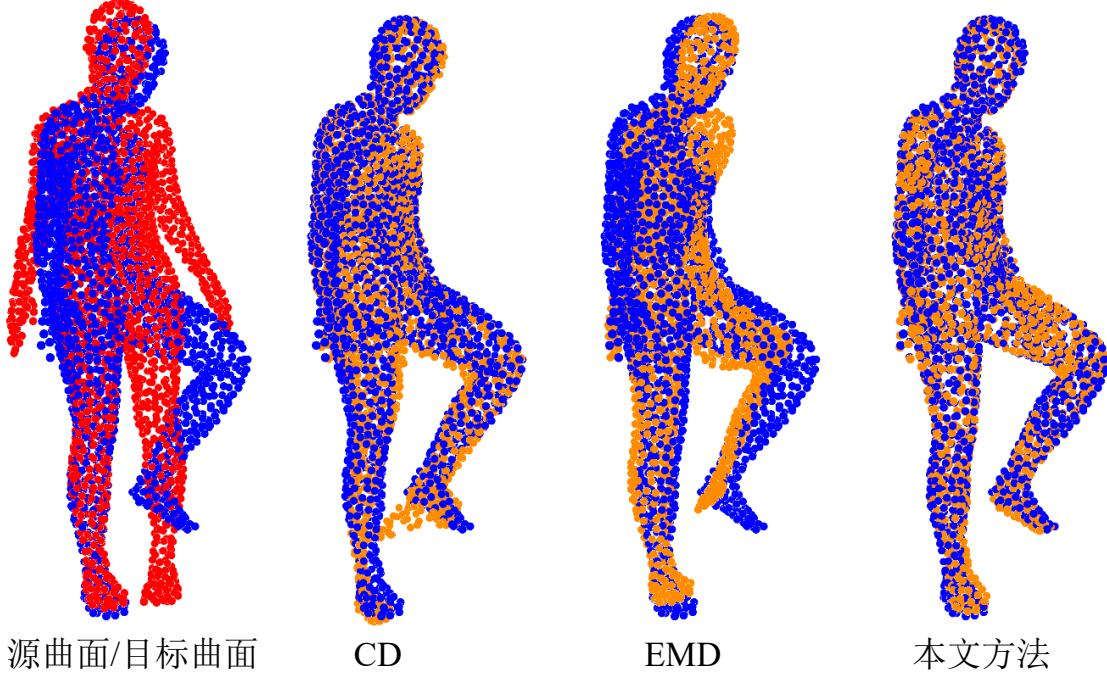


图 4.4 通过拟合一对样本比较不同的损失函数。上图用橙色的点云来表示形变后的结果。从源（红色）到目标（蓝色），CD 和 EMD 损失项都倾向于将左腿拉到右腿，而本文的损失可以成功驱动源点云变为目標形状。

表示可以用于深度学习的框架而已。此外，公式(4.6) 中的更新公式明显不同于 RAFT。

在第一次迭代之前，本文还使用附加的 DGCNN 从源点云中提取初始隐藏状态  $h_0$  和几何特征  $f_s$ 。在第  $k$  个阶段，本文将几何特征  $f_s$  和相关度  $C_{k-1}$  聚合在一起  $x_k = [C_{k-1}, f_s]$  作为更新单元的输入。GRU 中的全连接层被 MLP 取代。从  $x_k$  和  $h_{k-1}$ ，和<sup>[149]</sup>一样，本文用 GRU 获得更新的隐藏状态  $h_k$ 。然后， $h_k$  通过两个 MLP 来预测  $\mathbf{w}_k^k$  和  $\psi_k$ 。新的形变点云  $S^k$  可以根据公式(4.6)得到。

## (2) 损失函数

成功的无监督学习框架的关键是设计合适的损失函数。虽然倒角距离 (Chamfer Distance, CD) 和推土机距离 (Earth Movers' Distance, EMD) 是衡量曲面形状距离的常用指标，但它们依赖于搜索最近点或最优传输流，导致通过最小化 CD 或 EMD 来驱动源曲面变形的到目标曲面的时候，通常大概率会发生错误。一个例子如图4.4 所示，该样例在一对具有不同损失函数的模型上过拟合。CD 和 EMD 损失函数都无法将源点云准确地形变为目標点云。

本文不是直接搜索最近点或最佳传输流，而是基于形状相似度的度量构造损失函数<sup>[150-153]</sup>。类似于光场描述子 (Light Field Descriptor, LFD)<sup>[150]</sup>，本文将三维形状投影到多视角二维平面上，并通过投影的二维深度图像和掩码图像计算相似度，进一步的衡量变形点云和目标点云的差距。为了使整个过程可训练，本文设计了一种可微渲染方法，将点云渲染为二维深度图像和掩码图像。此外，

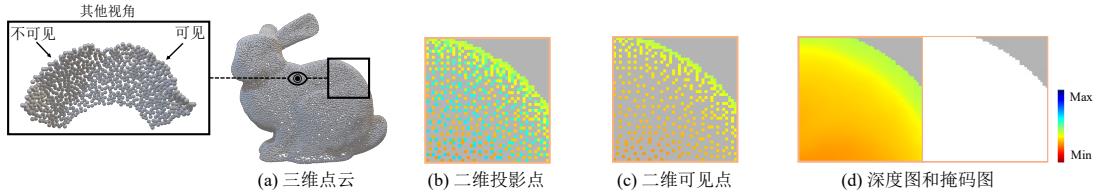


图 4.5 可微渲染过程的图示。(a): 输入点云。(b): 给定点云和相机, 本文将所有点投影到前视图, 并将深度图的值设置为投影点的 z 值。(c): 本文根据投影在以  $p_i$  为中心的窗口中的点的深度值去除像素  $p_i$  周围的不可见点。(d):  $p_i$  的深度值是通过窗口中投影的可见点 z 值的加权平均计算得到的, 本文顺带由此可以得到掩码图像。

本文为刚性变换的变量和蒙皮权重设计了正则化项。下面, 为方便起见, 本文使用  $\tilde{\mathcal{S}}$  来表示  $k$  阶段的形变形状  $\mathcal{S}^k$ 。

**深度损失项。**对于点云  $\mathcal{P}$  和给定的视角  $v$ , 本文将  $\mathcal{P}$  转换为相机坐标系  $\mathcal{P}_v$  并计算深度映射  $\mathcal{D}(\mathcal{P}_v)$ 。本文首先收集  $\mathcal{P}_v$  中的点, 其二维投影在  $\mathcal{D}(\mathcal{P}_v)$ , 并将这些点的集合定义为  $\mathcal{N}(p_i)$ 。然后,  $\mathcal{N}(p_i)$  的最小和最大 z 值表示为  $\min_i, \max_i$ 。如图4.5的(b) 和 (c) 所示, 本文从  $\mathcal{N}(p_i)$  中去掉 z 值超过  $(\min_i + \max_i)/2$  的点, 因为它们可能来自不可见部分, 本文将  $\mathcal{N}(p_i)$  的可见部分表示为  $\mathcal{V}(p_i)$ 。对于  $g_j \in \mathcal{V}(p_i)$ , 本文将权重  $w_{ij}$  设置为:

$$w_{ij} = \frac{\exp(-\rho_{ij}/\gamma)}{\sum_{g_m \in \mathcal{V}(p_i)} \exp(-\rho_{im}/\gamma)}, \quad (4.7)$$

其中  $\rho_{ij}$  表示  $p_i$  与  $g_j$  的二维投影之间的平方距离,  $\gamma$  控制深度图的锐度。这样,  $\mathcal{D}(\mathcal{P}_v)$  上的像素  $p_i$  的深度值  $d_i$  由  $\mathcal{V}(p_i)$  中点的 z 值的加权平均值计算得出:

$$d_i = \sum_{g_j \in \mathcal{V}(p_i)} w_{ij} g_j^z, \quad (4.8)$$

其中  $g_j^z$  表示点  $g_j$  的 z 值。

对于点云  $\tilde{\mathcal{S}}$  和  $\mathcal{T}$ , 本文将它们的深度计算为  $\mathcal{D}(\tilde{\mathcal{S}}_v)$  和  $\mathcal{D}(\mathcal{T}_v)$ 。这些成对深度图之间的损失定义为

$$\mathbf{L}_{\text{depth}}(\tilde{\mathcal{S}}, \mathcal{T}) = \mathbb{E}_{v \sim V} \|\mathcal{D}(\tilde{\mathcal{S}}_v) - \mathcal{D}(\mathcal{T}_v)\|_2^2, \quad (4.9)$$

其中  $V$  表示相机视角的集合。在反向传播期间, 深度图  $\mathcal{D}(\tilde{\mathcal{S}}_v)$  上像素  $p_i$  处的梯度  $\nabla d_i$  将通过公式(4.7) 中的  $w_{ij}$  影响  $\mathcal{V}(p_i)$ 。

**掩码损失项。**通过将三维曲面投影到二维平面, 本文还可以得到它的二维二进制掩码  $\mathcal{M}(\mathcal{P}_v)$ 。对于  $\mathcal{M}(\mathcal{P}_v)$  上的每个像素, 如果其与  $\mathcal{G}_v$  的投影的距离小于给定阈值, 则掩码值为 1. 否则值为 0。在计算  $\tilde{\mathcal{S}}$  和  $\mathcal{T}$  的掩码为  $\mathcal{M}(\tilde{\mathcal{S}}_v)$  和  $\mathcal{M}(\mathcal{T}_v)$ , 本文将掩码损失定义为:

$$\mathbf{L}_{\text{mask}}(\tilde{\mathcal{S}}, \mathcal{T}) = \mathbb{E}_{v \sim V} \|\mathcal{M}(\tilde{\mathcal{S}}_v) - \mathcal{M}(\mathcal{T}_v)\|_1, \quad (4.10)$$

全局掩码损失的反向传播过程可以通过以下方式计算。令  $c_i$  表示  $\mathcal{M}(\tilde{\mathcal{S}}_v)$  上像素  $p_i$  的值,  $\nabla c_i$  表示  $c_i$  的梯度。点  $\tilde{s}_j \in \tilde{\mathcal{S}}_v$  的梯度可以计算为:

$$\nabla \tilde{s}_j^z = \sum_{p_i \in \mathcal{M}(\tilde{\mathcal{S}}_v)} \nabla c_i \cdot \frac{\exp(-\tilde{\rho}_{ij}/\tilde{\gamma})}{\sum_{m=1}^M \exp(-\tilde{\rho}_{im}/\tilde{\gamma})},$$

其中  $\nabla \tilde{s}_j^z$  表示  $\tilde{s}_j$  的 z 坐标梯度,  $\tilde{\gamma}$  控制这个损失的锐度,  $r\tilde{\rho}_{ij}$  表示  $p_i$  与  $\tilde{s}_j$  的投影 (到  $x0y$  平面) 之间的平方距离。由于本文采用了软光栅化策略, 一个像素的掩码损失可以影响  $\tilde{\mathcal{S}}$  的所有点。

**刚性保持损失项。**形变点云的边长应通过以下项尽可能接近原始边长:

$$\mathbf{L}_{\text{arap}}(\tilde{\mathcal{S}}) = \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{E}} (\|\mathbf{p} - \mathbf{q}\|_2 - d_{ij})^2, \quad (4.11)$$

其中  $\mathcal{E}$  是在输入点云  $\mathcal{S}$  中根据邻居关系构造的边集,  $d_{ij}$  是输入点云中顶点对的距离。

**正则化损失项。**刚性变换  $\psi_k$  包括旋转矩阵  $\mathbf{R}_k$  和平移向量  $\mathbf{t}_k$ 。本文将平移向量的范数约束如下:

$$\mathbf{L}_{\text{tran}}(\mathbf{t}_k) = \|\mathbf{t}_k\|_2^2. \quad (4.12)$$

考虑到非刚性形变可能会像人体关节一样有跳跃, 本文对蒙皮权重添加一个稀疏项:

$$\mathbf{L}_{\text{sparse}}(\mathbf{w}_k^k) = \|\mathbf{w}_k^k\|_1. \quad (4.13)$$

第  $k$  个阶段的总损失由上述所有项构成:

$$\begin{aligned} \mathbf{L}^k = & \mathbf{L}_{\text{depth}}(\mathcal{S}^k, \mathcal{T}) + \beta_1 \mathbf{L}_{\text{mask}}(\mathcal{S}^k, \mathcal{T}) \\ & + \beta_2 \mathbf{L}_{\text{arap}}(\mathcal{S}^k) + \beta_3 \mathbf{L}_{\text{tran}}(\mathbf{t}_k) + \beta_4 \mathbf{L}_{\text{sparse}}(\mathbf{w}_k^k) \end{aligned} \quad (4.14)$$

最终的损失函数是所有阶段的组合:

$$\mathbf{L} = \sum_{i=1}^K \gamma^{K-i} \mathbf{L}^i, \quad (4.15)$$

其中  $\gamma \leq 1$  使得各个阶段的权重呈指数增长。

### 4.3 实验结果和讨论

在本节中, 本文给出了实现细节、消融实验、结果和比较。

### 4.3.1 实现细节

**数据集。**本文首先在一个融合数据集上进行了训练和测试，该数据集包括四种可形变物体，包括穿衣人体、裸身人体、猫和狗。对于穿衣人体数据，本文使用 HumanMotion<sup>[154]</sup> 数据集，其中包含 10 个人体运动序列。本文随机抽取 37600 个训练对和 1652 个测试对。对于裸身人体数据，本文使用 SURREAL<sup>[155]</sup> 数据集，该数据集由 68036 个视频组成，其中包含以 SMPL<sup>[156]</sup> 表示的合成人体模型。本文分别提取 65000 和 3036 个点云对用于训练和测试。猫和狗来自 TOSCA<sup>[157]</sup> 数据集。因为数据量不大（11 个猫和 9 个狗），本文通过 ACAP<sup>[158]</sup> 插值构造大量数据。对于训练，本文构建了 20604 个狗的点云对和 32270 个猫的点云对。为了测试，本文构建了 1500 个狗的点云对和 1500 个猫的点云对。总的来说，本文在这个实验中使用了 155474 个训练对和 7688 个测试对。本文还在 FaceWareHouse 数据集<sup>[36]</sup> 上进行了测试，该数据集包含 150 个不同个体的脸型和 47 种不同的表情。本文随机选择 20000 和 500 个人脸对进行训练和测试，其中每一对源点云和目标点云是随机选择的两个人的不同表情。本文还在 DFAUST<sup>[159]</sup> 数据集（2732 个训练对和 100 个测试对）中测试了本文的方法，其中包含自然噪声、异常值和不完整性。此外，为了验证本文的模型可以推广到刚性配准任务，本文在 ModelNet40 数据集<sup>[147]</sup> 上训练了本文网络的刚性版本。本文将每个类别分成 9:1 的比例用于训练和测试。每个提取的点云首先去中心化，然后缩放到半径为 0.5 的球体中。为了构建源点云和目标点云对，本文使用  $[0, 45^\circ]$  范围内的随机旋转角度和  $[-0.5, 0.5]$  范围内的平移。

**实验设定。**对于可形变物体和人脸的非刚性配准实验，提取的点数分别为 2048 和 5334。对于原始扫描数据，本文采样 2048 个控制点输入网络，然后通过径向基函数插值形变整个原始扫描模型。对于刚性配准实验，点数为 1024。对于 FaceWareHouse 数据，本文从原始拓扑中裁剪正面，并直接将 5334 个顶点作为点云。在非刚性配准实验中，公式(4.14) 中每一项的权重设置为 0.1, 0.01, 0.1, 10 和  $\gamma = 1.0$ 。对于刚性配准，本文只使用前两项， $\beta_1 = 0.1$  和  $\gamma = 0.8$ 。本文采用热启动策略来训练模型。一开始，网络只用 1 个循环阶段进行训练。每进行  $5K$  次训练迭代，本文将循环阶段的数量增加 1，直到阶段数量达到 7 个。在三个实验中，训练迭代的总数分别为  $100K$ 、 $50K$  和  $50K$ ，并且批大小都是 4。所有训练和测试均在具有 32 个 Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz、128GB RAM 和四个 32G V100 GPU 的工作站上进行。对于具有不同输入点点数 1024、2048 和 5334 的输入点云，7 个循环阶段的推理时间分别需要 0.14s、0.25s 和 0.55s。多视角损失项的相机视角是从单位球体的球坐标中采样的，如图4.1所示。

**评估指标。**本文使用 CD 和 EMD 评估配准性能。对于 FaceWareHouse 数据集，本

损失函数	# 视角	# 阶段	CD	EMD
CD	-	7	1.652	4.962
EMD	-	7	3.241	4.542
深度	$5^2$	7	0.701	0.431
深度 + 掩码	$5^2$	7	0.691	0.426
深度 + 掩码	$7^2$	7	0.618	0.396
深度 + 掩码	$9^2$	7	0.600	0.386
深度 + 掩码	$11^2$	3	10.717	12.404
深度 + 掩码	$11^2$	4	4.571	5.234
深度 + 掩码	$11^2$	5	2.329	3.116
深度 + 掩码	$11^2$	6	1.110	0.951
深度 + 掩码	$11^2$	7	<b>0.599</b>	<b>0.386</b>

表 4.1 消融实验的结果，指标为 CD( $\times 10^{-4}$ ) 和 EMD( $\times 10^{-3}$ )。

文进一步计算逐点均方误差 (MSE)，因为它们共享相同的拓扑。对于刚性配准，本文使用均方根误差 (RMSE) 和平均绝对误差 (MAE) 评估转换，如 [82]。

### 4.3.2 消融实验

本文进行消融实验以证明每个组件的重要性。具体来说，本文使用四类可形变物体的数据集作为这部分的基准。消融实验是针对损失函数、视图数量和循环阶段的数量而设计的。表4.1 展示定量结果。如第二行所示，CD 损失函数和 EMD 损失函数的配准误差仍然很大，即使它已经在不同参数下尝试过的最佳结果。通过本文提出的深度损失，配准结果显著改善（第 3 行）。额外的掩码损失（第 4 行）带来了进一步的性能提升。本文也逐渐增加视图的数量（第 4,5,6,11 行），最后选择  $11^2$  个视图。将视图数量固定为  $11^2$ ，本文在第 7-11 行中展示不同循环阶段的结果。可以观察到，随着更多的迭代，配准精度继续变得更好。为了平衡配准精度和计算速度，本文最终使用了 7 个阶段。

### 4.3.3 结果和比较

#### (1) 在可形变物体上进行配准

本文与经典优化方法 CPD<sup>[85]</sup>、其最近改进的版本 BCPD<sup>[89]</sup> 和基于学习的方法 CPD-Net<sup>[90]</sup> 和 PR-Net<sup>[98]</sup> 进行比较。

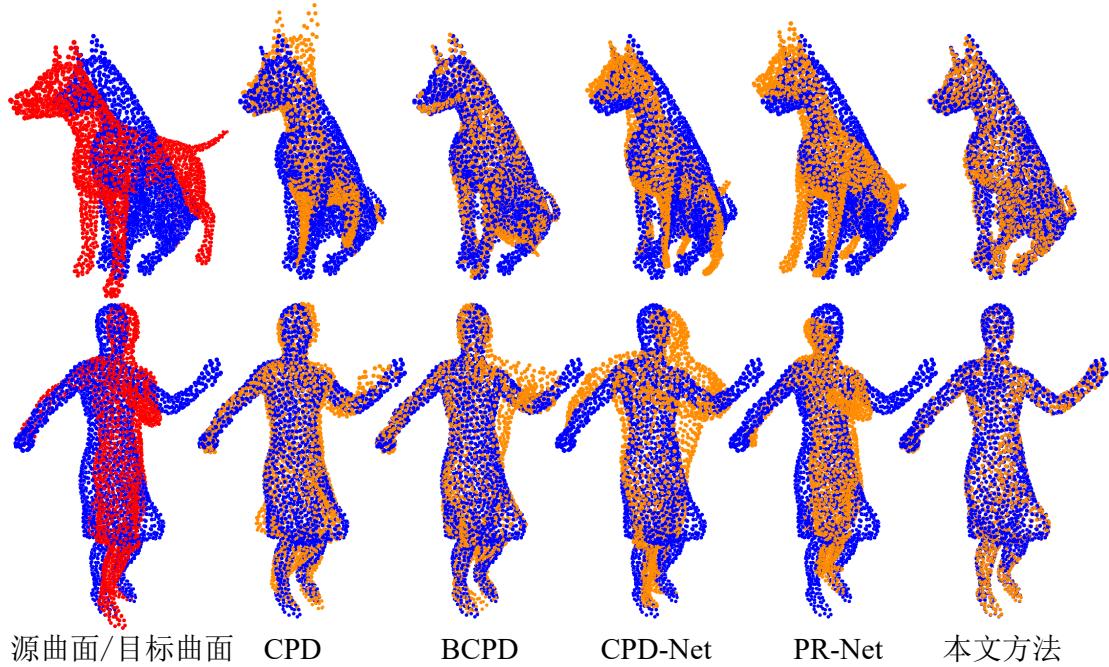


图 4.6 在可形变物体上的比较。源点云、目标点云和形变点云分别用红色、蓝色和橙色表示。

数据集	度量	源点云	CPD	BCPD	CPD-Net	PR-Net	本文方法
可形变物体	CD	37.246	4.126	2.375	14.678	29.457	<b>0.599</b>
	EMD	25.952	7.853	5.478	21.696	25.192	<b>0.386</b>
FaceWareHouse	EMD	1.230	1.168	0.979	1.054	1.304	<b>0.578</b>
	MSE	21.469	9.568	8.013	13.752	14.575	<b>5.245</b>

表 4.2 可形变物体数据集的指标: CD( $\times 10^{-4}$ ) 和 EMD( $\times 10^{-3}$ )。FaceWareHouse 数据集的指标 EMD( $\times 10^{-2}$ ) 和 MSE( $\times 10^{-4}$ )。

表4.2 展示了不同方法在可形变物体和 FaceWareHouse 数据集上的性能。图4.6展示了定性比较。本文的方法在质量和数量上都显著优于以前的方法。基于优化的方法 CPD 和 BCPD 无法处理大规模的非刚性形变，因此它们在具有挑战性的测试集上表现不佳。CPD-Net 也无法处理形变，这应该是由于逐点偏移场的大自由度造成的。虽然 PR-Net 降低了自由度，但基于样条的表示不能很好地表达形变，导致结果不佳。相比之下，由于本文精心设计的非刚性表示、网络结构和损失项，本文的方法表现最好。

## (2) 人脸配准

除了在粗尺度非刚性形变物体上进行测试外，本文还对细尺度形变物体进行了测试和与其他方法的比较。本文尝试将具有表情的人的一种脸型配准到具有不同表情的另一个人的脸。考虑到本文只处理正面，本文在这个实验中把所有

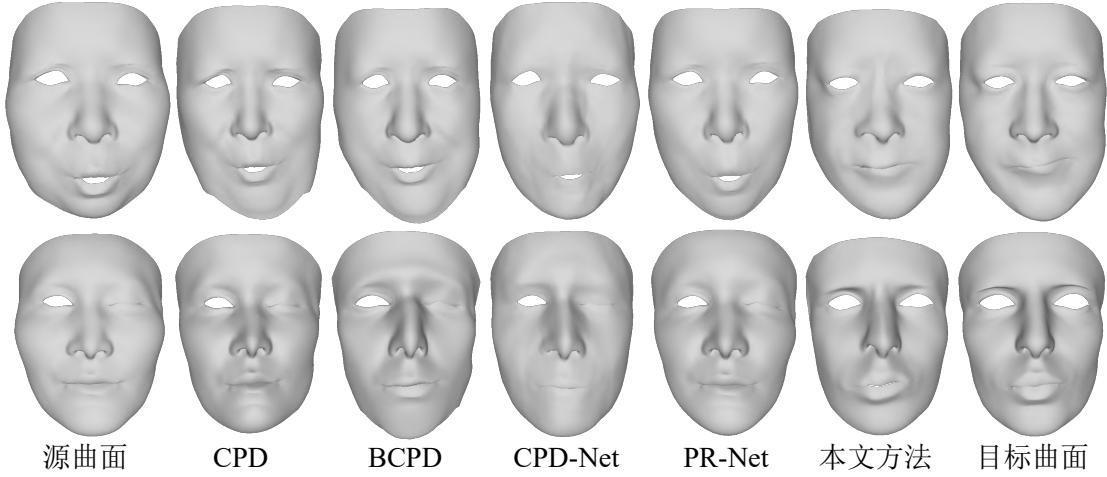


图 4.7 在 FaceWareHouse 数据集上的比较。为了更好地可视化，本文将网格的面片也渲染了出来。

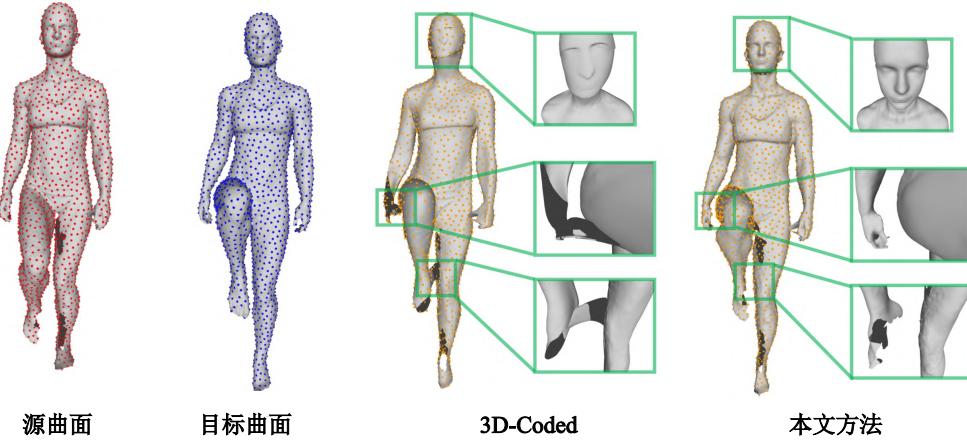


图 4.8 在 DFAUST 数据集上，与 3D-Coded 进行比较。

的相机视图都放在了人脸的正面。

表4.2展示了不同方法在 FaceWareHouse 数据集上的性能。图4.7展示了定性比较。本文的方法比所有以前的方法表现得更好，不仅在 EMD 和 MSE 指标上，而且在配准结果的视觉质量上。在图4.7的第一行中，本文的方法是唯一能够将张嘴形变为闭合的方法。此外，在第二行中，本文的结果是唯一可以让眼睛睁开的结果，表明本文的方法比以前的方法具有更好的捕捉高精度人脸形变的能力。

### (3) 原始扫描数据的配准

本文还在原始扫描数据集 DFaust<sup>[159]</sup> 上进行训练和测试，并与 3D-Coded<sup>[160]</sup> 方法进行比较。图4.8展示了比较结果。测试集上的源点云、3D-Coded 结果和本文的结果与目标曲面间的平均 CD ( $\times 10^{-4}$ ) 分别为 9.02、0.43、0.24。尽管该数据集中包含一些缺陷，但本文的方法仍然取得了令人满意的结果，并且比 3D-Coded<sup>[160]</sup> 更好地保留了几何结构和细节。该实验表明，本文的方法仍然适用于真实的扫描数据。

#### (4) 刚性配准

本文的完整框架也可以轻松扩展以执行刚性配准任务。在本实验中，本文通过预测每个阶段的单个刚性变换（如4.2.2中所讨论的）将本文的网络转换为刚性版本，并在 ModelNet40 数据集上测试性能。

比较方法包括基于局部优化的方法 ICP<sup>[67]</sup>、两种基于全局优化的方法 Go-ICP<sup>[76]</sup> 和 FGR<sup>[161]</sup>，以及基于学习的方法 DCP<sup>[82]</sup>。DCP<sup>[82]</sup> 由旋转和平移的真实值监督，并在本文自己的网络中使用相同的数据集进行训练。为了与 DCP 进行公平比较，本文还训练了 RMA-Net 的另一个变体，它使用真实值作为监督，除了以无监督方式训练模型（与之前的实验相同）。本文的模型经过 7 个阶段的训练，但可以在推理时与任意阶段一起使用。在本文的实验中，本文在测试期间使用了 10 个阶段。表4.3 展示了不同方法在 ModelNet40 数据集上的刚性配准性能。可以观察到，本文的方法在刚性情况下比以前的方法表现更好，这证明了本文提出的框架的普遍适用性。

度量	RMSE(R)	MAE(R)	RMSE(t)	MAE(t)
ICP <sup>[67]</sup>	19.041	7.585	0.133	0.154
Go-ICP <sup>[76]</sup>	13.086	1.891	0.060	0.026
FGR <sup>[161]</sup>	10.143	1.928	0.048	0.030
DCP <sup>[82]</sup>	2.057	1.313	0.013	0.023
本文方法（无监督）	1.287	0.344	0.008	<b>0.007</b>
本文方法（有监督）	<b>0.735</b>	<b>0.265</b>	<b>0.006</b>	0.009

表 4.3 在 ModelNet40 数据集上的比较。

#### (5) 更多结果与分析

**蒙皮权重的分布。**为了更好地观察形变曲面上的权重分布，本文选择了一些样本并在图4.9 中可视化了蒙皮权重。首先展示每个阶段的形变结果点云。其中，本文将阶段 2 到 7 的权重可视化 ( $\{\mathbf{w}_k^k\}, k=2, \dots, 7$ )，将每个  $\mathbf{w}_k^k$  的绝对值归一化到  $[0, 1]$  范围内，并根据颜色条渲染点。从图4.9 可以看出，蒙皮权重的分布平滑且局部。平滑度有助于生成的点云保持合理的形状，权重分布的局部性验证了本文的循环策略降低了每个阶段的自由度。此外，可以观察到，权重倾向于集中在形变相对较大的部分，这也解释了本文方法对大规模形变的有效性。

**不同阶段的结果。**为了更清楚地展示本文的循环策略的效果，本文在每个阶段可视化了可形变物体数据集中的一些样本。如图4.10所示，形变点云在循环过程中越来越接近目标点云。

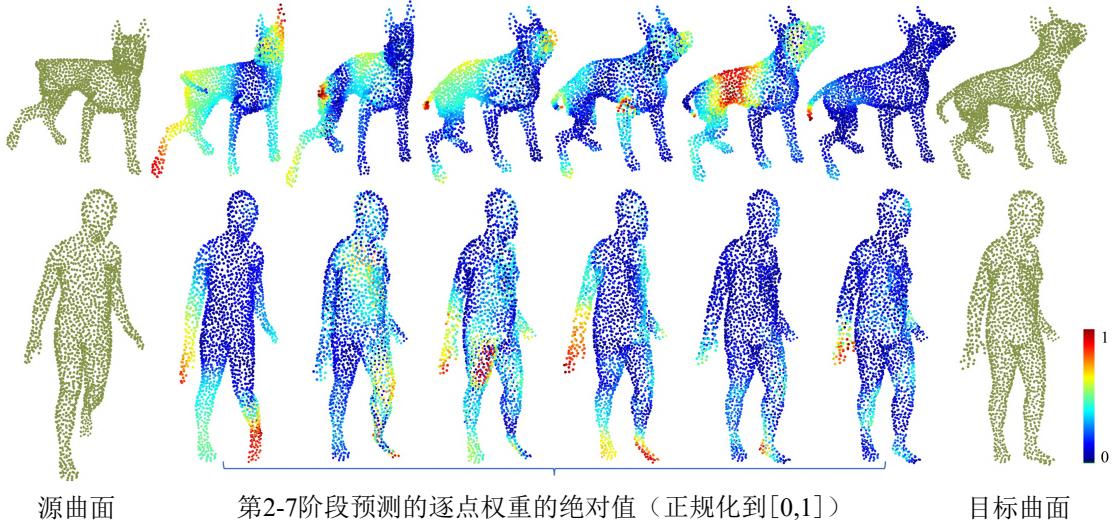


图 4.9 形变曲面上的权重分布。

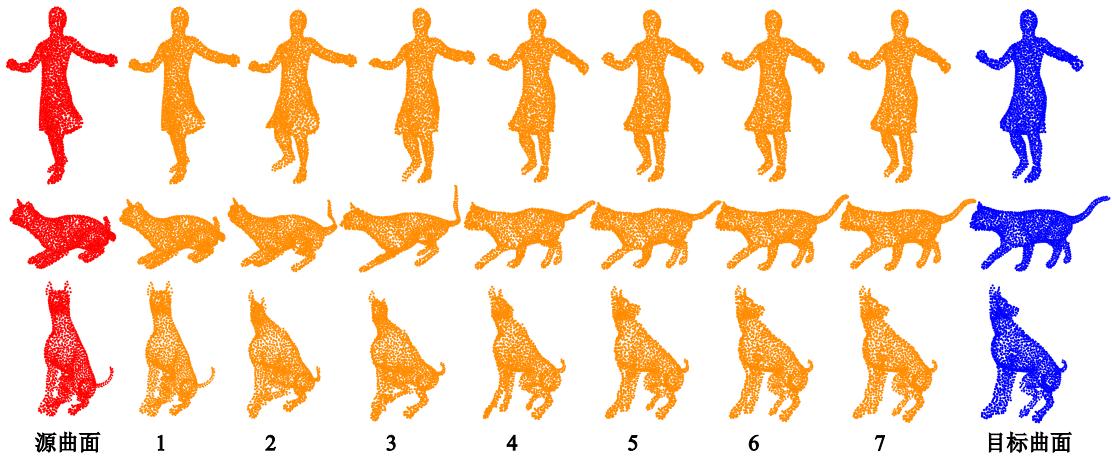


图 4.10 不同阶段的结果。

**更多样本的比较。**本文还展示了更多样本以将本文的方法与 CPD<sup>[85]</sup>、BCPD<sup>[89]</sup>、CPD-Net<sup>[90]</sup> 和 PR-Net<sup>[98]</sup>进行比较。在图4.12 中，本文展示了可形变物体数据集中更多的比较样本。在图4.11 中，本文展示了 FaceWareHouse<sup>[36]</sup> 数据集中的更多比较样本。

#### 4.4 本章小结

本文提出了一种新的非刚性表示，可以通过循环网络进行学习。进而，本文也提出了相应的网络结构，RMA-Net，一种将本文提出的表示用于非刚性配准的无监督学习网络。其次，本文设计了一个多视角对齐损失函数来指导网络训练，而无需真实值对应作为监督。广泛的消融实验已经验证了本文完整框架中每个组件的有效性。本文还优于以前最先进的非刚性配准方法，证明了本文提出的方法的优越性。

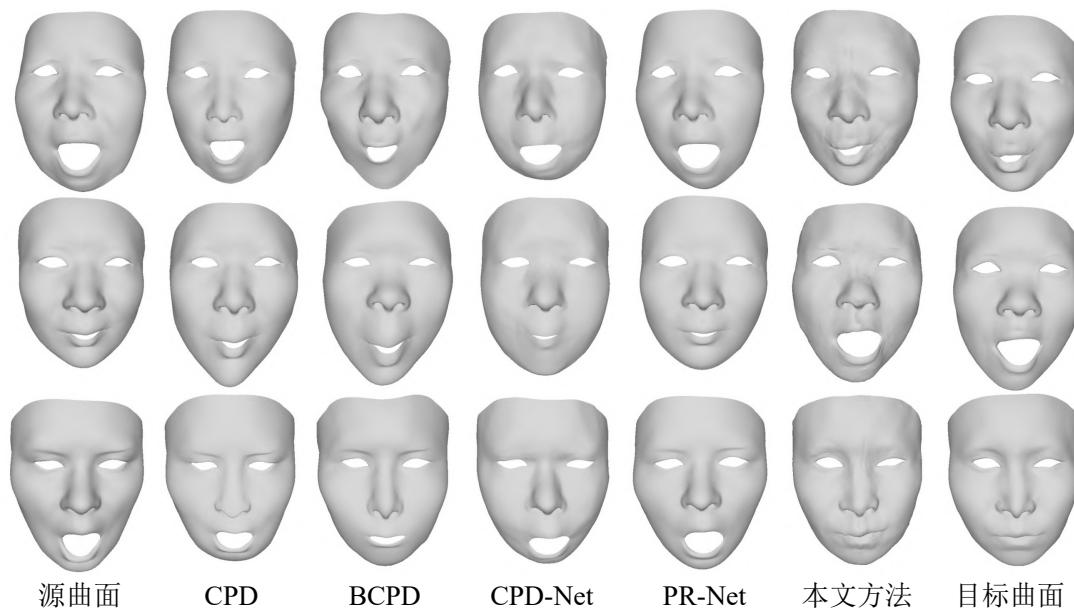


图 4.11 更多 FaceWareHouse 数据集中测试样本的比较。

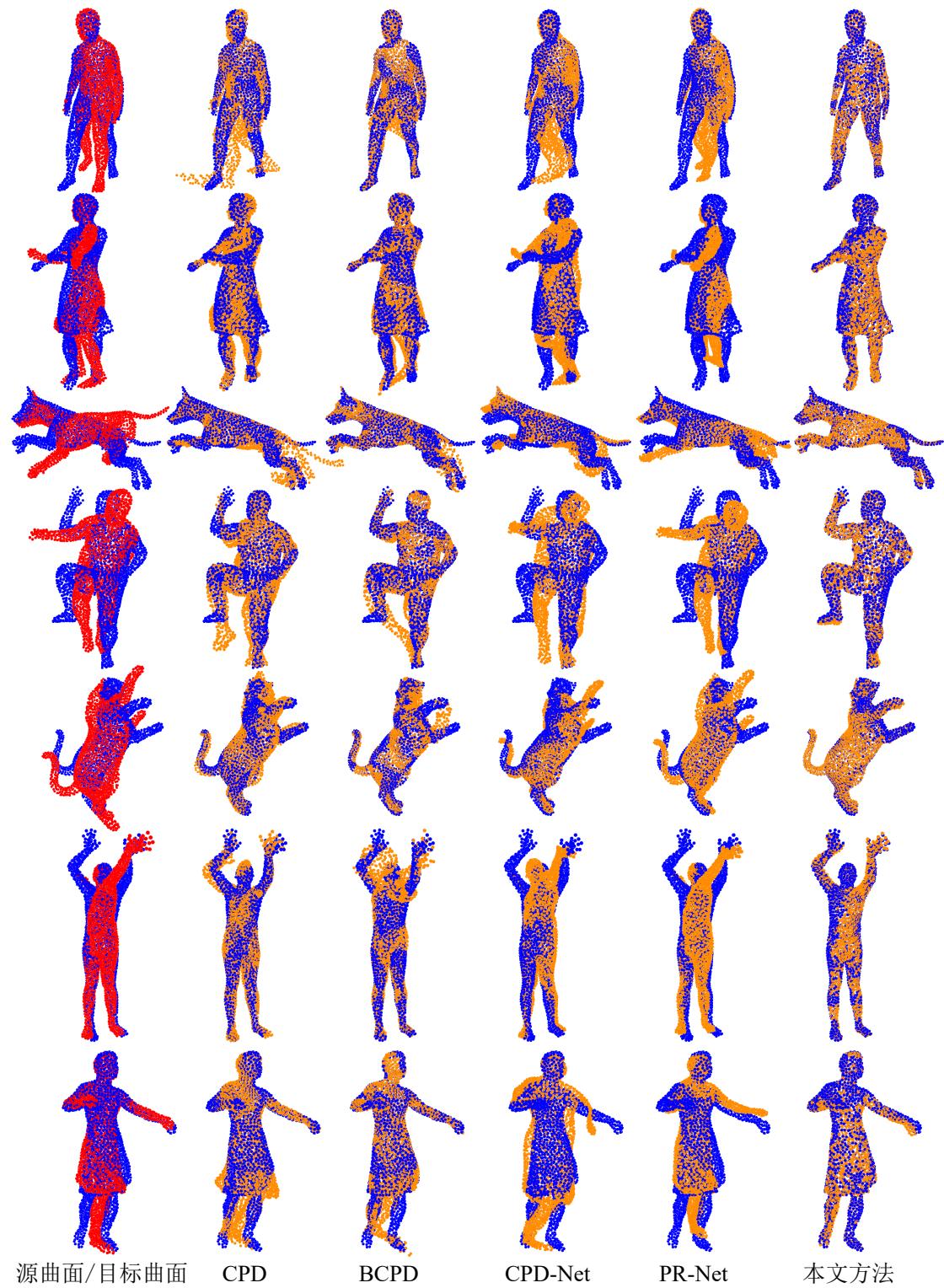


图 4.12 更多可形变物体数据集中测试样本的比较。

# 第5章 基于可微形变节点图的非刚性形变表示

## 5.1 引言

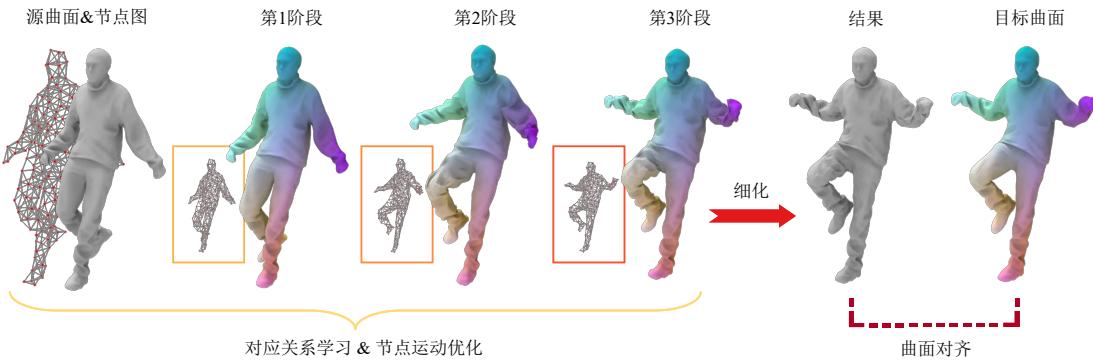


图 5.1 本文提出了一种基于可微形变节点图的非刚性形变表示。整体流程采用由粗到精的方式，利用节点图表示进行结构级配准，然后利用顶点级变形来精修几何细节。对于结构级配准，与传统的基于优化的方法相比，本文用神经学习模块代替了对应关系构造、点对拒绝和迭代策略。对于顶点级配准，本文在最终阶段中应用逐点细化模块以进一步改善配准结果。

配准问题是计算机视觉、机器人和图形学中一个相当基本的问题，可以用于融合不同的时空观测结果以重建全局完整的曲面形状。其在跟踪<sup>[138-139]</sup>、重建<sup>[33-34]</sup>等诸多任务中都发挥着关键作用。尽管该问题已经研究了多年，但如何有效且高效地进行配准仍然是一个具有挑战性的问题，因为它本质上是一个组合优化问题。与刚性配准相比，非刚性配准<sup>[67,85,90-91]</sup>是由于其较大的自由度而更具挑战性，并且对于具有部分重叠和缺失区域的数据会更加糟糕。

曲面配准任务广泛采用的算法流程<sup>[67,85]</sup>通常迭代地进行对应关系预测步骤和形变对齐步骤。对应关系预测步骤构建源曲面和目标曲面之间的逐点对应关系。通过构建的对应关系，形变对齐步骤通过解优化问题来估计空间变换，该优化问题使源曲面变形以在某些度量下更接近目标曲面。这两个步骤交替应用，直到收敛。最终的配准结果在很大程度上依赖于构造对应的质量，这并非易事<sup>[144,162-166]</sup>。在传统的基于优化的方法中，最近邻搜索等启发式对应关系预测策略，以及简单的迭代方式可能会导致局部最优，使得算法缺乏鲁棒性。

最近，基于学习的方法取得了长足的进步，并且表现出了更好的鲁棒性<sup>[80,82,90-91]</sup>。然而，这些方法主要是为刚性配准而开发的。与刚性情况相比，基于学习的非刚性曲面配准发展较少。非刚性配准的一个挑战是如何表示非刚性变形，因为它具有高自由度。逐点位移已应用于非刚性度较小的情况（例如基于样条的扰动<sup>[90]</sup>，场景流<sup>[91]</sup>等）。本文在上一章中提出的方法 RMA-Net<sup>[167]</sup>提出了使用刚性变换逐点加权组合的方式迭代地进行非刚性形变，该方案采用了一种新的学习的策略。另一方面，基于节点图<sup>[99]</sup>的表示通常用于非刚性配准，

因为它具有降低复杂度和能够利用几何形状先验的优点。然而，由于节点数量、边连接和拓扑可能因形状不同而不同，直接利用节点图作为网络训练的输入并不容易。

本章节提出了一种基于可微形变节点图的非刚性形变表示，并且这种表示适用于神经网络方法。该方法通过用基于神经的策略替换一些组件来充分利用训练好的神经网络中嵌入的形状先验和领域知识。第一个替换是对应关系预测模块。本文设计了一个神经网络来预测对应关系，而不是使用人为设计的特征（例如最近邻）。本文还通过减少对低质量点对的权重来预测逐点可靠性置信度。第二个关键模块是迭代学习策略。本文采用了一个基于门循环单元（Gated Recurrent Unit, GRU）<sup>[148]</sup>的循环结构，它保留了更多的迭代阶段的历史记忆，以取代传统的迭代方式。因此，本文应用分层正则化策略来控制每次迭代的自由度，从而降低训练这个循环网络的难度。最后，本文添加了一个精细化模块，在逐点级别对变形形状进行精细化，消除了由于节点图表示能力有限而导致的误差。整个框架以自监督的方式进行端到端的训练。大量实验表明，本文提出的方法在合成数据和实际扫描数据上都优于之前最先进的方法。

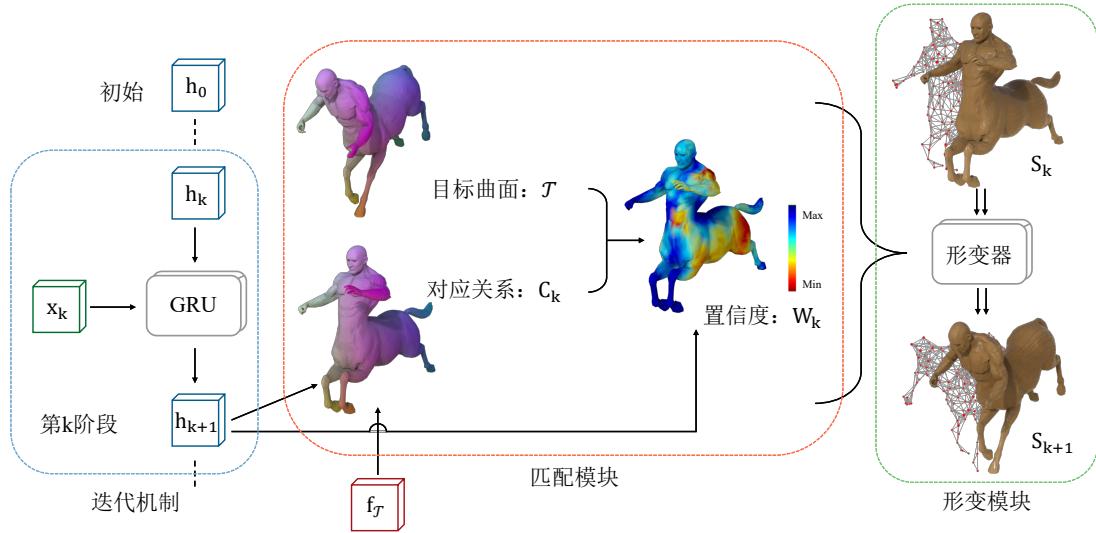


图 5.2 本文的网络结构。本文采用基于 GRU 的循环框架，并在每个迭代阶段操作匹配和变形模块。在匹配模块中，网络预测了具有逐点置信度的软对应。在变形模块中，本文使用可微分形变器来优化非刚性形变，使变形后的形状更接近目标曲面。

## 5.2 基于可微形变节点图的非刚性形变表示：设计与应用

给定源曲面和目标曲面，本文将其表示为点集  $S \in \mathbb{R}^{M \times 3}$  和  $T \in \mathbb{R}^{N \times 3}$ 。任务的目标是估计一个非刚性变形形状  $\tilde{S} \in \mathbb{R}^{M \times 3}$ ，使得  $\tilde{S}$  与目标曲面  $T$  很好地对齐。在本节中，本文将首先介绍本文的算法流程，然后给出本文的损失函数设计。

### 5.2.1 算法流程

为了解决配准问题，流行的算法流程是以迭代方式交替预测对应关系和求解变形，广泛用于基于优化的配准方法<sup>[100-101,168-169]</sup>中。令下标  $k$  表示第  $k$  次迭代阶段，并表示  $\mathcal{S}_0 = \mathcal{S}$ ，则迭代过程可以概括为：

$$\begin{aligned} (\mathcal{S}_k, \mathcal{T}) &\xrightarrow{\text{Match}} \mathbf{C}_k, \\ (\mathcal{S}_k, \mathcal{T}, \mathbf{C}_k) &\xrightarrow{\text{Deform}} \mathcal{S}_{k+1}, \end{aligned} \quad (5.1)$$

其中匹配过程构造对应关系  $\mathbf{C}$ ，变形过程通常通过求解优化问题来估计变形。对于变形表示，节点图<sup>[99]</sup>是一种流行的选择，受益于低自由度和良好的结构级形状先验。

在本项工作中，本文参考了上述经典算法流程并改进了一些模块，以形成更鲁棒的框架。本文还利用节点图表示来实现令人满意的结构级配准结果。然而，由于节点图的自由度小，可能会丢失一些几何细节，这促使了算法在最终采用顶点级的细化模块，从而形成整体从粗到细的框架。对于结构级配准，本文重新设计了对应关系构造和网络迭代的方式，与当前使用的启发式策略（如对应构造中的最近邻搜索）相比，可以获得更好的解决方案。本文的算法流程概览如图5.1所示。

整个框架由一个基于 GRU<sup>[148]</sup> 的循环网络进行组织，其中每个阶段包括一个匹配模块和一个变形模块，如图5.2 所示。

**匹配模块。**在每次迭代中，本文预测源曲面和目标曲面之间的稠密对应关系，表示为一个由非负元素组成的软匹配矩阵： $\mathbf{C} \in \mathbb{R}^{M \times N}$ 。 $\mathbf{C}$  的每一行总和为 1，其中的元素表示源曲面中的点对应于目标曲面上的每个点的概率。为了获得预测，本文提取目标的嵌入特征（表示为  $f_{\mathcal{T}}$ ）并将其应用于每次迭代。在第  $k$  次迭代中，对应映射  $\mathbf{C}_k$  计算为：

$$\mathbf{C}_k = \text{Softmax}(\langle h_k \cdot f_{\mathcal{T}} \rangle), \quad (5.2)$$

其中  $h_k$  表示循环网络中的隐藏状态，可以记住每个阶段的历史特征。本文首先计算  $h_k$  和  $f_{\mathcal{T}}$  的内积，然后对每一行使用 Softmax 归一化。通过软对应，本文可以形成从源曲面到目标曲面的软投影： $\mathcal{S} \rightarrow \mathbf{C}_k \mathcal{T}$ ，其中  $\mathcal{S}$  中的每个点与目标曲面上的点软相关。根据  $\mathcal{S}$ 、 $\mathbf{C}_k \mathcal{T}$  和  $h_k$ ，网络随后预测这些对应关系的置信度  $\mathbf{W}_k \in (0, 1)^M$ ，用来表示所预测的对应关系的可靠性。

**变形模块。**一旦获得预测的匹配矩阵  $\mathbf{C}_k$  和置信度  $\mathbf{W}_k$ ，本文就可以使用它们来

求解变形。这里的变形定义在  $\mathcal{S}$  的节点图上, 记为  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{M})$ , 其中

$$\begin{aligned}\mathcal{V} &= \{v_i \in \mathbb{R}^3, i = 1, \dots, m\}, \\ \mathcal{E} &= \{(v_i, v_j), v_i \text{ 相邻于 } v_j\}, \\ \mathcal{M} &= \{(R_i, t_i) \in SE(3), i = 1, \dots, m\},\end{aligned}\tag{5.3}$$

分别表示节点集、边集和节点刚性运动集。通过在  $\mathcal{G}$  上定义运动  $\mathcal{M}$ , 源曲面  $\mathcal{S}$  可以变形为目标曲面  $\mathcal{M} \circ \mathcal{S}$ , 这里本文使用符号  $\circ$  来表示变形操作。对于给定的  $\mathcal{S}$  和  $\mathcal{V}$ , 变形结果  $\mathcal{M} \circ \mathcal{S}$  完全由运动  $\mathcal{M}$  决定。

具体的变形过程解释如下。本文参考之前的工作<sup>[99]</sup>来制定变形函数。对于  $\mathcal{S}$  中的点  $p \in \mathbb{R}^3$ , 本文首先利用测地线距离找到它的  $L$  个最近节点  $\{u_j\}_{j=1}^L \subseteq \mathcal{V}$ , 其中本文在实验中设置  $L = 8$ 。然后, 本文又通过以下方式来计算蒙皮权重  $\{\alpha_j\}_{j=1}^L$ :

$$\begin{aligned}\eta_j &= \left(1 - \frac{\|p - u_j\|}{d_{max}}\right)^2, \\ \alpha_j &= \eta_j / \sum_{j=1}^L \eta_j,\end{aligned}\tag{5.4}$$

其中  $d_{max}$  表示  $p$  与其  $L + 1$  个最近节点的距离,  $\|\cdot\|$  表示测地距离。那么变形函数可以表示为:

$$\tilde{p} = \sum_{j=1}^L \alpha_j (R_j(p - u_j) + u_j + t_j),\tag{5.5}$$

其中  $\tilde{p}$  表示变形曲面上与源曲面中的  $p$  相关的点。使用公式(5.4)和公式(5.5), 整个源曲面可以根据本文框架的变形模块中获得的图形运动  $\mathcal{M}$  进行变形。

用于变形的运动是通过求解优化问题获得的。在第一阶段,  $\mathcal{M}_1$  被初始化为恒等变换。当  $k > 1$  时,  $\mathcal{M}_k$  被初始化为  $\mathcal{M}_{k-1}$ 。为了优化, 本文使用了两个能量项。首先, 变形的结果应该与预测的对应一致。具体来说, 本文鼓励变形结果接近于匹配模块中计算的  $\mathbf{C}_k \mathcal{T}$ , 相关能量项可以写成:

$$\mathbb{E}_{deform}(\mathcal{M}_k) = \|\mathbf{W}_k \odot (\mathcal{M}_k \circ \mathcal{S} - \mathbf{C}_k \mathcal{T})\|_F^2.\tag{5.6}$$

其中  $\odot$  表示逐点乘法。此外, 本文还在  $\mathcal{G}$  上求解  $\mathcal{M}_k$  时使用了正则化能量项。就像在<sup>[99]</sup> 中一样, 对于每一对邻居, 本文将应用于邻居的变换与实际变换的邻居位置之间的平方距离相加, 鼓励邻居节点上的变换是一致的:

$$\mathbb{E}_{reg}(\mathcal{M}_k) = \sum_{(v_i, v_j) \in \mathcal{E}} \|R_i(v_j - v_i) + (v_i + t_i) - (v_j + t_j)\|_2^2,\tag{5.7}$$

其中  $R_i$ 、 $t_i$  和  $t_j$  表示在  $\mathcal{M}_k$  的相邻节点  $v_i$  和  $v_j$  上定义的变换。因此，本文可以通过以下方式求解最优运动  $\tilde{\mathcal{M}}$ ：

$$\tilde{\mathcal{M}}_k = \arg \min_{\mathcal{M}_k} (\mathbb{E}_{deform}(\mathcal{M}_k) + \lambda_k \cdot \mathbb{E}_{reg}(\mathcal{M}_k)) \quad (5.8)$$

为了解决优化问题，本文参考<sup>[45]</sup>并使用高斯-牛顿方法，保持变形模块可微。总的来说，变形模块的公式可以写成：

$$\mathcal{S}_k = \tilde{\mathcal{M}}_k \circ \mathcal{S}. \quad (5.9)$$

在这里，需要注意到公式(5.8) 中的  $\lambda_k$  是最终变形曲面  $\mathcal{S}_k$  的自然正则化。当  $\lambda_k \rightarrow \infty$  时，变形后的  $\mathcal{S}_k$  应该接近  $\mathcal{S}$  的刚性变换。基于这一观察，本文采用分层正则化策略：

$$\lambda_1 > \lambda_2 > \dots > \lambda_K \approx 0, \quad (5.10)$$

这将使网络逐步学习变形，降低每一步的难度，提升配准效果。

**迭代机制。**不同于简单的迭代方案（直接使用当前结果作为下一次迭代的输入，如公式(5.1)），本文采用基于 GRU 的循环网络进行迭代。采用固定更新格式<sup>[149,167,170]</sup>，在第  $k$  个阶段，GRU 将当前特征  $x_k$  的组合作为输入，并将隐藏状态从  $h_{k-1}$  更新为  $h_k$ 。对于  $x_k$  的计算和隐藏状态  $h_0$  的初始化，本文与之前的工作<sup>[149,167]</sup>保持一致。以下过程的计算基于更新的隐藏状态  $h_k$  和目标  $f_{\mathcal{T}}$  的嵌入特征。在交替迭代期间，本文将  $h_k$  和  $f_{\mathcal{T}}$  传递到下一阶段的匹配和变形模块中。

在最后一次迭代之后，网络仍然需要学习结构级变形和目标曲面之间的几何细节层面上的差距。具体来说，本文使用了一个细化模块，它从  $h_K$  和  $f_{\mathcal{T}}$  预测逐点位移以实现顶点级配准，表示为  $\mathcal{S}'$ 。这个模块也很容易训练，因为变形曲面和目标曲面之间的距离在早期的迭代中已经显著减少。

### 5.2.2 损失函数

对于真实扫描的可形变物体，通常非常难获得标记良好的数据。为了应对这种局面，本文和上一章中提出的方法一样，以无监督的方式训练网络。在本节中，本文介绍用于网络训练的损失项。为了描述方便，本文首先考虑单次迭代，其中预测的匹配矩阵和变形曲面分别表示为  $C$  和  $\tilde{\mathcal{S}}$ 。

**倒角距离损失项。**为了使源曲面变形以使变形曲面  $\tilde{\mathcal{S}}$  与目标  $\mathcal{T}$  整体对齐，本文采用倒角距离衡量他们的距离：

$$\mathcal{L}_{cad}(\tilde{\mathcal{S}}, \mathcal{T}) = \frac{1}{2|\tilde{\mathcal{S}}|} \sum_{\tilde{s} \in \tilde{\mathcal{S}}} \min_{t \in \mathcal{T}} \|\tilde{s} - t\|_2^2 + \frac{1}{2|\mathcal{T}|} \sum_{t \in \mathcal{T}} \min_{\tilde{s} \in \tilde{\mathcal{S}}} \|\tilde{s} - t\|_2^2. \quad (5.11)$$

**刚性保持损失项。**为了鼓励预测的对应关系是平滑的，本文在这里使用刚性保持损失项，尽可能鼓励形变是刚性的。具体地，本文鼓励邻居间的距离在  $\mathbf{C}\mathcal{T}$  和源曲面之间得到保持。设下标  $(i), (j)$  表示行选择操作，刚性保持损失项定义为：

$$\mathcal{L}_{arap}(\mathbf{C}) = \sum_{(i,j) \in \mathcal{E}_S} (\|\mathbf{C}_{(i)}\mathcal{T} - \mathbf{C}_{(j)}\mathcal{T}\|_2 - \|S_{(i)} - S_{(j)}\|_2)^2, \quad (5.12)$$

其中  $\mathcal{E}_S$  和  $\mathcal{E}_{\mathcal{T}}$  表示  $S$  和  $\mathcal{T}$ 。

**置信度损失项。**为了防止置信度  $\mathbf{W}$  降为 0，本文鼓励  $\mathbf{W}$  的值不要太小：

$$\mathcal{L}_{conf}(\mathbf{W}) = -\|\mathbf{W}\|_2^2, \quad (5.13)$$

其中  $\mathcal{E}_S$  和  $\mathcal{E}_{\mathcal{T}}$  表示  $S$  和  $\mathcal{T}$ 。

假设本文使用  $K$  次迭代进行训练，第  $k$  次迭代中的损失可以表示为：

$$\mathcal{L}_k = \mathcal{L}_{cd}(S_k, \mathcal{T}) + \beta_1 \mathcal{L}_{arap}(\mathbf{C}_k) + \beta_2 \mathcal{L}_{conf}(\mathbf{W}_k). \quad (5.14)$$

**细化损失。**尽管基于节点图的表示在保持形状结构方面表现得非常好，但它的自由度很小，因此不能很好地变形几何细节。因此，本文在算法的最后添加了一个细化模块，并鼓励细化曲面  $S'$  接近目标曲面。此外，本文限制细化位移不要太大。考虑到这些方面，细化曲面  $S'$  的损失函数定义为：

$$\mathcal{L}_{refine}(S') = \mathcal{L}_{cd}(S', \mathcal{T}) + \epsilon \|S' - S_K\|_2^2. \quad (5.15)$$

总损失可以写成：

$$\mathcal{L}_{sum} = \sum_{k=1}^K \gamma^{K-k} \mathcal{L}_k + \mathcal{L}_{refine}. \quad (5.16)$$

## 5.3 实验结果和讨论

在本节中，本文进行了广泛的实验来分析算法组件，并展示了在实际扫描数据上的非刚性形变结果和比较，证明了本文方法的优越性。

### 5.3.1 实现细节

**数据集。**本文在合成数据和真实扫描数据上训练和测试本文的模型，包括 6 类可变形物体：人体、猫、狗、狼、半人马和马。对于合成数据，本文将本文的模型应用在 TOSCA<sup>[157]</sup> 数据集中，该数据集有几类合成动物模型。本文还在真实扫描数据集 Dynamic FAUST<sup>[159]</sup> 中训练和测试本文的模型，其中包含 10 个原始扫描人体曲面序列。从 TOSCA 数据集中，本文选择了 16551 个训练对和 513 个测试对。从人类序列中，本文总共选择了 7,802 个训练对和 391 个测试对。真实

扫描的数据缺乏每一对中源和目标之间的真值对应关系，模型包含随机噪声以及真实扫描过程中的不完整性，增加了配准的难度。此外，本文还在 *Kinect Azure DK* 获得的更多有缺陷真实数据上测试了本文训练的模型，以展示模型的鲁棒性。

**实验设置。**对于所有训练和测试曲面，本文采样 4096 个点以形成源点云和目标点云。对于所有模型，本文构建了节点数在 150 – 200 之间的节点图。在高斯-牛顿优化过程中，本文使用 5 次迭代，迭代步长为 1.0、0.8、0.7、0.6、0.5。该网络包含 3 次迭代，其中  $\{\lambda_k\}_{k=1}^3$  分别为  $10^3$ 、 $10^2$ 、10。公式(5.14) 中的  $\beta_1$  和  $\beta_2$  设置为 5 和  $10^{-4}$ ，公式(5.15) 中的  $\epsilon$  设置为 0.03，公式(5.16) 中的  $\gamma$  设置为 0.9。本文用 OneCycleLR<sup>[171]</sup> 策略，循环长度为 1000，最大学习率为  $10^{-4}$ ，批大小为 4。该网络使用 Adam<sup>[172]</sup> 优化器完全训练了 500000 次前向-后向迭代。所有实验均在具有 40 个 Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GHz、128GB RAM 和四个 24G GeForce RTX 3090Ti GPU 的工作站上进行。

**评估指标。**对于合成数据和实际扫描数据的结果，本文使用 CD 和 EMD 评估了配准的性能，和在<sup>[167]</sup> 中相同。

### 5.3.2 消融实验

本文通过实验分析了网络关键设置的选择，包括迭代次数、细化模块、逐点置信度。之后本文又分析了变形节点数所带来的影响。

# 迭代次数	$\{\lambda_k\}$	细化	置信度	CD ↓	EMD ↓
1	$\{10^3\}$	×	×	22.34	9.32
2	$\{10^3, 10^2\}$	×	×	13.36	8.02
3	$\{10^3, 10^2, 10\}$	×	×	6.33	5.20
4	$\{10^3, 10^2, 10, 1\}$	×	×	7.23	5.78
3	$\{10^3, 10^2, 10\}$	√	×	5.10	4.23
3	$\{10^3, 10^2, 10\}$	√	√	<b>4.84</b>	<b>4.09</b>

**表 5.1 网络设置的消融实验结果，由 CD( $\times 10^{-5}$ ) 和 EMD( $\times 10^{-3}$ ) 作为评估度量。**

**网络设定。**为了探索最佳网络设置，本文研究了迭代次数、细化模块和对应置信度。使用 Dynamic FAUST<sup>[159]</sup> 数据集作为基准，本文应用不同的策略对相同的数据和相同的训练设置进行训练和测试，如表 5.1。

起初，本文只使用 1 次迭代，没有细化模块或对应置信度（展示在第 1 行中）。然后本文逐步添加迭代次数。从 1 – 4 行中，迭代次数的最佳选择是 3，正

则化权重分别为  $10^3$ 、 $10^2$  和 10。第 4 行的结果不如第 3 行的结果，因为太小的正则化权重增加了自由度并导致训练困难。因此，本文在所有实验中都使用 3 次迭代。在第 5 行中，本文在最后一次迭代后添加细化模块，发现误差再次减小，展示细化过程的效果。最后，本文在第 6 行中添加了对应置信度，这也促进了配准效果。这也符合本文的预期，因为置信度提供了一种自然的软对拒绝策略，这在优化方法中也很重要。在本文对合成数据集和真实扫描数据集的主要实验中，本文保持网络设置始终与第 6 行相同。

**变形节点数。**构造节点图时，需要手动调整节点数。本文应用实验来搜索最有利于网络训练的节点数。本文从 TOSCA<sup>[157]</sup> 数据集中收集了 30 对形状，并为它们创建了 10 个版本的节点图，其中节点数量逐渐增加。

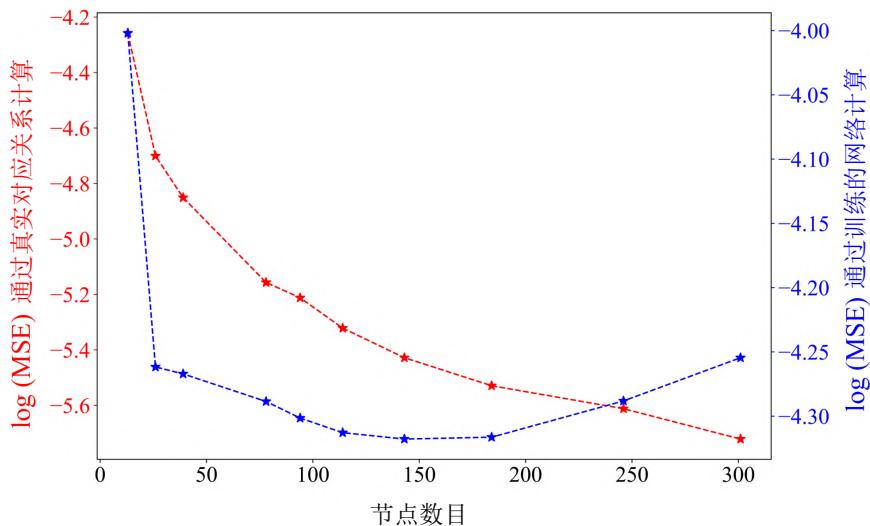


图 5.3 图的节点数分析。红线表示使用真实值进行优化的误差。蓝线代表无监督训练的误差。

首先，本文直接使用真实值对应来解高斯-牛顿优化 ( $\lambda = 10$ )。变形后的形状与目标形状的 MSE 如图 5.3 中的红线所示，其中横轴表示节点数，纵轴（左侧，红色）代表 MSE。然后本文应用本文的网络（1 次迭代，无监督）在这个集合上进行足够的前向-后向迭代训练（本文将所有数据使用 1000 遍以确保收敛），并将拟合误差展示为图 5.3 中的蓝线，其中横轴表示节点数，纵轴（右侧，蓝色）表示网络结果的误差。

虽然更多的节点可以通过使用真实值进行优化来驱动更小的拟合误差，但本文无监督训练的最佳选择是相对适中的节点数量。节点太少的图缺乏表达能力，节点太多的图自由度太大，不利于本文的无监督网络训练。

### 5.3.3 结果和比较

在本节中，本文展示了在合成数据集 TOSCA 和真实扫描数据集 Dynamic FAUST 上与最近最先进的方法的定性和定量结果和比较，展示了本文方法的优

越性。虽然 CPD<sup>[85]</sup> 是最经典和最著名的非刚性配准优化方法，但 BCPD<sup>[89]</sup> 是 CPD 的最新贝叶斯版本，在收敛并在优化方法中达到最先进的效果。最近的另一项工作，RMA-Net<sup>[167]</sup>，是一个基于学习的框架，将非刚性变形参数化为一系列刚性变换的组合，它也是经过训练的以无监督的方式实现的。之前的一些工作<sup>[160,173]</sup> 考虑了可变形形状和模板形状之间的配准，但本文不与它们进行比较，因为本文不假设本文的任务中有预设的模板。此外，本文还与一个专注于估计曲面对应关系的框架进行了比较。具体来说，本文与 FMNet<sup>[174]</sup> 进行比较，这是一个基于泛函映射的有监督网络。预测的稠密对应可以将每个源点拉到其对应位置，可以看作是对应关系定义的自然变形。

### (1) 合成数据上的配准

本文首先在合成数据集上训练和测试本文的模型，并将其与 CPD、BCPD、RMA-Net 和 FMNet 进行比较。从 TOSCA 数据集中，本文选择了 5 类动物来组成训练和测试集，包括猫、狗、马、半人马和狼。表 5.2 展示了每种方法在 TOSCA 数据集上的性能比较。可以看到，本文的方法获得了最好的性能。

度量	源模型	CPD	BCPD	FMNet	RMA-Net	本文方法
CD↓		47.50	27.29	16.31	59.9	3.12
EMD↓		27.76	4.43	3.05	16.25	0.45

表 5.2 TOSCA 数据集上的结果和比较，指标为 CD( $\times 10^{-5}$ ) 和 EMD( $\times 10^{-3}$ )。

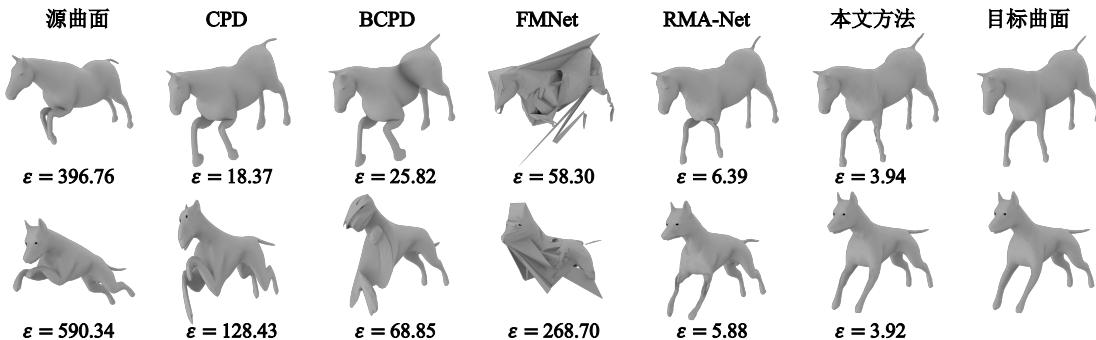


图 5.4 与基于优化的方法 CPD<sup>[85]</sup>、BCPD<sup>[89]</sup> 和基于学习的方法 FMNet<sup>[174]</sup>、RMA-Net<sup>[167]</sup> 在合成数据集 TOSCA<sup>[157]</sup> 上的比较。

基于优化的方法 CPD 和 BCPD 对特定的测试样本很敏感，可能无法收敛到正确的解，导致结果不够好。FMNet 预测稠密对应，通常通过测地线误差阈值下的准确度来评估。在本文估计变形的配准问题中，本文尝试了直接将 FMNet 的预测对应关系视为变形的简单方法。其低质量的结果表明，从稠密对应关系预测任务直接到良好的配准质量仍然存在一些差距，这些差距在本文的算法流程中通过一些专门设计的模块和端到端可训练框架得到了避免。

RMA-Net 基本可以将四肢拉到正确的位置（例如图5.4中的狗的腿），但是在某些细节上可能存在局部塌陷，这在本文的方法中由于节点图的正则化作用没有出现。此外，RMA-Net 可能会导致对拓扑的错误识别（例如图5.4 中的马，左大腿的一些点被拉到身体上），这在本文的结果中也不会出现，原因应该是节点图对形状有着潜在的暗示作用，从而可以降低拓扑识别错误的概率。

从图5.4 和表5.2，可以得出结论，本文的方法在非刚性曲面配准任务上效果很好。本文的方法充分利用节点图的形状先验和配准效果，其效果优于以前的方法。

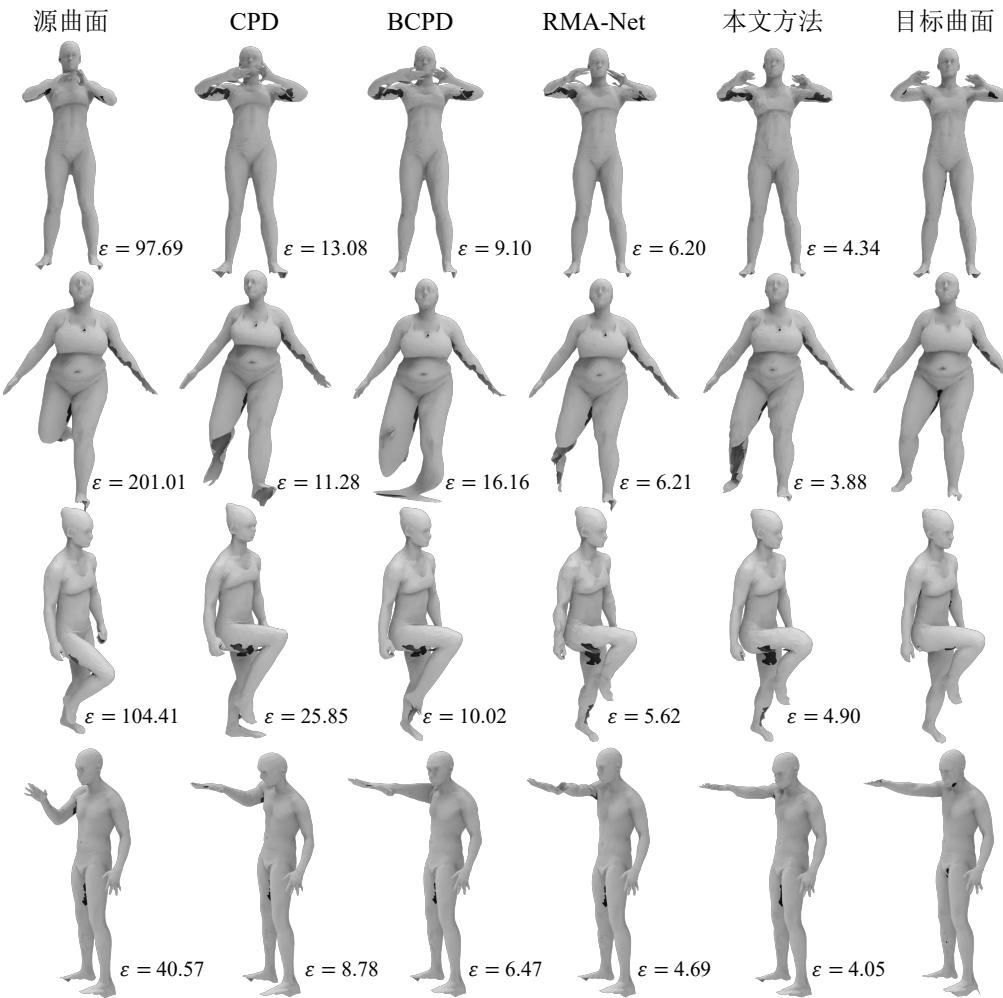


图 5.5 在实际扫描数据集 Dynamic FAUST 上与 CPD<sup>[85]</sup>、BCPD<sup>[89]</sup> 和 RMA-Net<sup>[167]</sup> 的比较。

## (2) 真实扫描数据配准

为了证明本文对真实扫描数据的可行性和鲁棒性，本文在 Dynamic FAUST 数据集上训练和测试本文的模型，其中真实扫描数据存在噪声、异常值和不完整性。本文与 CPD、BCPD 和 RMA-Net 的比较展示在图5.5 和表5.3 中。比较结论与合成数据的情景一致。本文的方法在定性和定量实验中都取得了最佳性能。

度量	Input	CPD	BCPD	RMA-Net	本文方法
CD↓	87.68	21.02	11.19	5.03	<b>4.84</b>
EMD↓	13.09	8.23	6.78	4.17	<b>4.09</b>

表 5.3 在 Dynamic FAUST 数据集的结果和比较，指标为 CD( $\times 10^{-5}$ ) 和 EMD( $\times 10^{-3}$ )。

在图5.5的第一行中，本文的结果是唯一将手拉开到正确目标位置的结果，而其他方法都没有将它们拉开。在第二行，CPD、RMA-Net 和本文的结果成功地将腿拉下，但只有本文的结果保持了小腿和脚的原始形状。第三个和第四个例子分别抬起腿和伸展手臂，本文的方法估计了正确的身体运动并保持整体曲面处于合理的形状，避免拉长小腿（第三排中的 CPD、BCPD）或弯曲手臂（第四行中的 CPD、RMA-Net）。从这些示例中，可以看到本文的方法仍然适用于低质量的实际扫描数据。

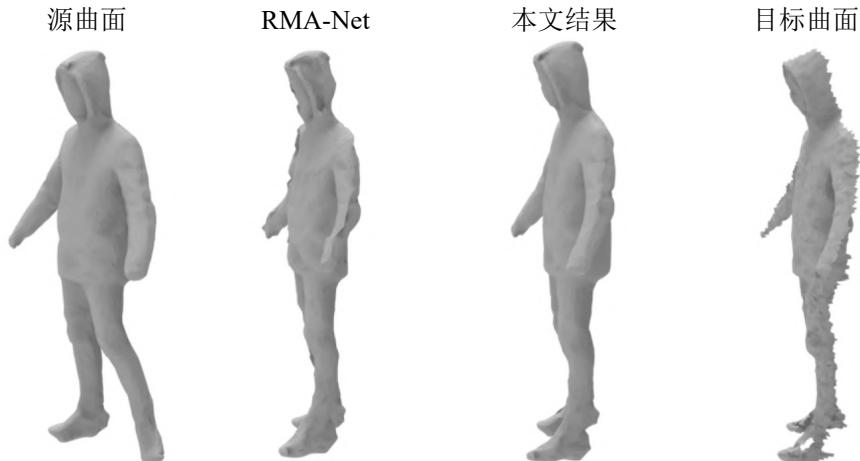


图 5.6 在由 *Kinect Azure DK* 收集的深度图像上和 RMA-Net<sup>[167]</sup> 的比较。

### (3) 更多缺陷数据的配准

为了进一步展示框架的鲁棒性，本文对 *Kinect Azure DK* 获取的一些深度图像进行了进一步测试。本文使用一个 4D 动态重建问题<sup>[31]</sup>所用的数据集，其中包含不同人的 14 个深度序列（本文构建了 4680 和 121 个对用于训练和测试）并尝试从完整的人体模型配准到深度图像。本文还在此数据集上和 RMA-Net 进行比较。结果如图5.6所示。可以看到，本文的结果在改变姿态时可以保持源模型的基本形状，而 RMA-Net 的结果不能保持源模型的基本形状。这展示了本文框架的鲁棒性。

### (4) 更多结果与分析

**各个阶段迭代结果。**图5.7列出了所有迭代阶段的迭代结果。从源曲面和其节点图，本文的框架在每次迭代中预测对应关系和置信度，然后根据预测的匹配信息

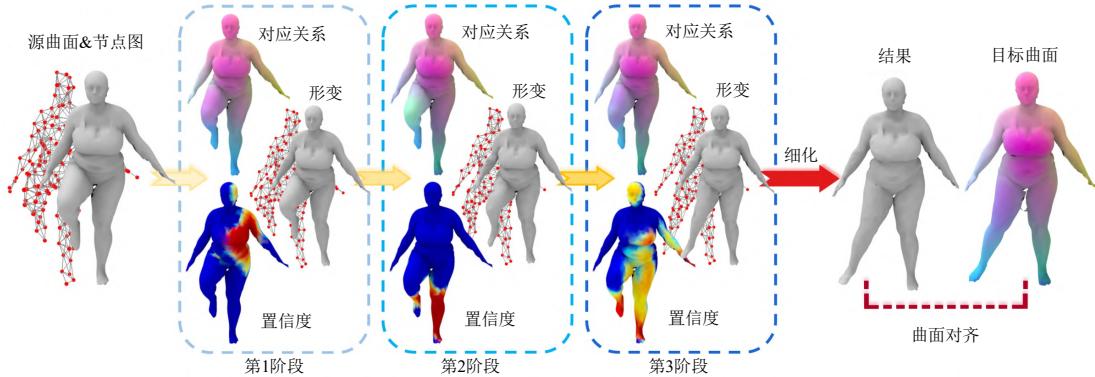


图 5.7 所有迭代阶段的结果，包括所有迭代中的对应关系、置信度、变形形状和节点图，以及最终细化的形状。

将形状与节点图一起变形。从第 1 个阶段到第 3 个阶段，可以看到对应关系越来越可靠，变形后的形状也越来越接近目标形状。最终通过细化阶段，配准结果在结构和顶点级别都与目标曲面很好地对齐了。

**更多结果。**图 5.8 列出了更多关于真实扫描数据集 Dynamic FAUST<sup>[159]</sup> 的结果。为了方便观察真实扫描曲面的质量，本文使用一些红色虚线圆圈来标记源网格和目标网格上的低质量区域。可以看到扫描的曲面存在噪声、异常值和不完整性。此外，每个源和目标对之间的质量缺陷不一致（比如，第三行中的源的左手断了，与目标不同），进一步增加了配准难度。第一列和第三列展示源曲面和目标曲面，第二列展示本文的配准结果。为了更好地观察配准效果，本文将结果和目标点一起渲染。可以看到，即使配准结果与源曲面具有相同的质量缺陷，结果形状也与目标形状很好地对齐，这表明了本文框架对实际扫描数据的鲁棒性。

## 5.4 本章小结

在这项工作中，本文提出了一种基于可微形变节点图的非刚性形变表示，将在传统优化中广泛使用的节点图方法转变成了神经网络版本。本文方法以由粗到精的方式进行非刚性配准，依次进行结构级别和顶点级别的配准。对于粗结构层次，本文沿用传统的基于节点图的算法流程，通过基于学习的策略改进了一些关键模块，包括对应关系构造和迭代机制。对于顶点级别，本文利用逐点细化模块来获得更好的几何细节。该网络以无监督的方式进行端到端的训练，并且在合成和实际扫描数据上都优于以前的最先进方法。

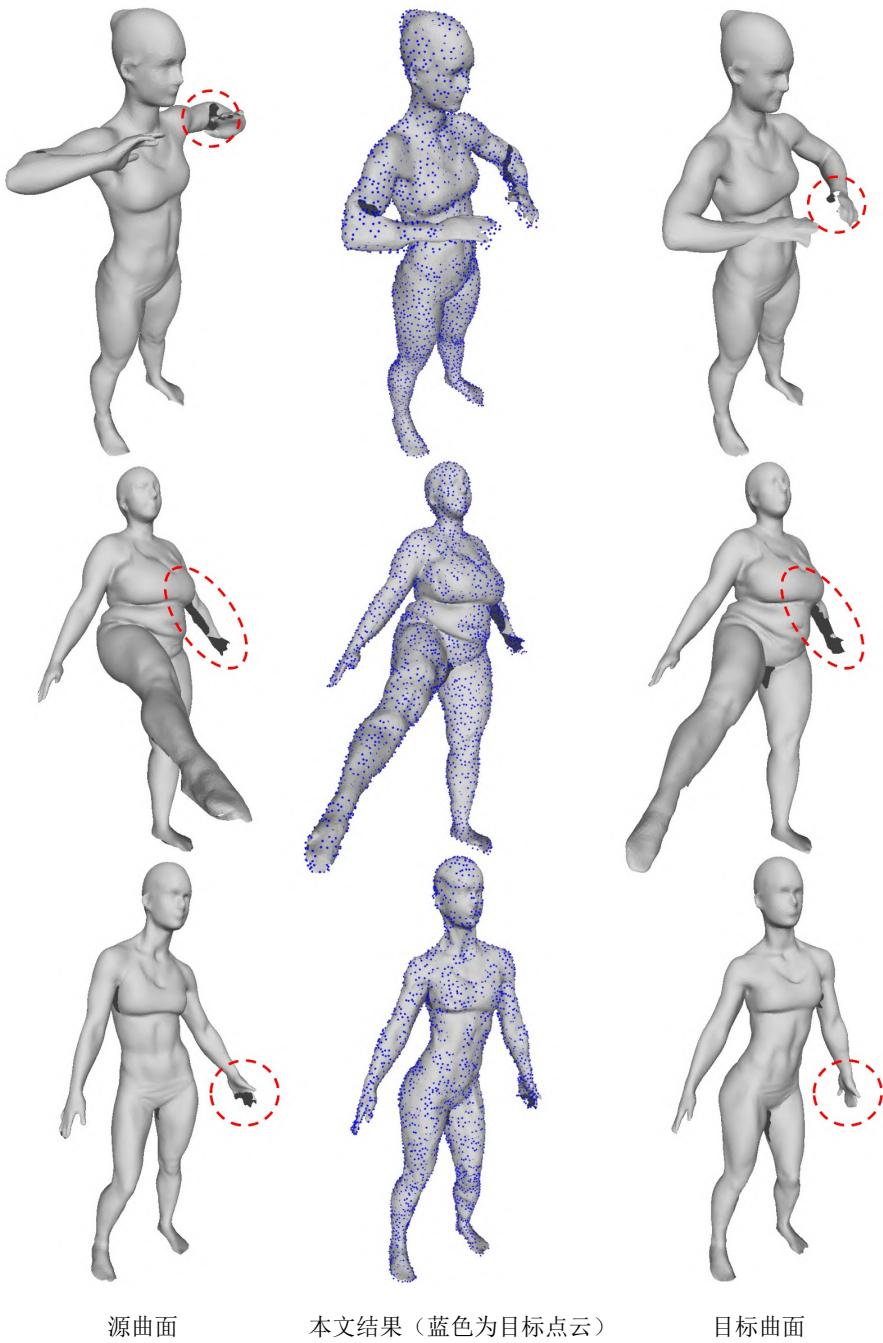


图 5.8 更多关于真实扫描数据集 Dynamic FAUST<sup>[159]</sup> 的结果。图中用红色虚线圆圈标记了源曲面和目标曲面的一些低质量部分，并将本文的结果和目标点一起呈现在第二列中。

## 第6章 总结与展望

### 6.1 本文总结

本篇论文针对三维几何领域的重要课题，三维几何表示，进行了研究。首先本文是针对单个三维物体，提出了两种新颖的表示。本文提出了基于神经场的点云局部隐式表示，打破了传统点云中将点云表示为离散三维点的方式，建立了连续的三维曲面，且具有泛化性，最终在点云上采样的任务当中，取得了比之前方法都要好的效果。本文也提出了基于径向基函数的局部形状表示，这种表示可以很好地被使用在具有高复杂度、丰富细节的三维几何模型上面，相较于之前的整体、显式的表示来说，有了非常大的提升，并且在三维模型几何细节修复的任务上取得了比之前的方法更好的效果。之后本文针对具有潜在对应关系的多个物体，提出了两种新的非刚性形变表示。本文首先提出了基于逐点加权刚性形变的非刚性几何形变表示，并成功将这种表示用在了神经网络当中，提升了非刚性配准的效果。之后，进一步地，本文将之前在优化方法当中广泛使用的节点图使用在了神经网络当中，构造了基于可微形变节点图的非刚性形变表示，利用节点图的结构化先验信息，进一步提升了非刚性配准的效果。

### 6.2 未来展望

正如第一章中所划分的，本文将未来展望分为单个物体的几何形状表示和多物体间的几何形变表示两个方面来分析。

#### 6.2.1 单个物体的几何形状表示未来展望

对于单个物体的几何形状表示来说，在不同的场景下，由于三维数据获取设备的不同以及目标任务的不同，所以导致最适合的三维形状表示也不同。也就是说不会出现一种统一的最好的形状表示。在未来，一个比较可观的技术发展路线是，对于每种现有的几何形状表示我们都去克服它内在蕴含的一些问题。

- 对于点云表示来说，本文已经在第二章当中，使用了基于神经场的点云局部隐式表示来克服了分辨率的问题，这克服了点云本质上的离散化的问题。未来的研究者或许可以在连续曲面的基础上提出更高的表示要求，比如高频特征或尖锐特征等。同时，也可以有更多的信息包含进来，从而实现更多的功能，如纹理上采样从而提高点云渲染的效果等等。
- 对于网格来说，尽管本文在第三章中提出的基于径向基函数的局部形状表示可以解决其高复杂度和非规整的问题，但是从当前关于网格的研究现状

来看，其和神经网络的结合还是未来可以突破的难点。在深度学习的框架中，如何可控地、可微地进行网格的生成，会是未来的重要研究方向。

- 对于深度图像来说，首先，如何克服采集设备的噪声所带来的影响是一个很有趣的研究方向。受制于原始深度数据的质量问题，发展基于学习的去噪和超分辨率方法会是很有意义的。此外，未来的研究者也可以致力于发展更高质量的深度图像采集设备，以便提供更好的三维几何信息。
- 对于多视角几何表示来说，正如第一章中所提到的，随着神经辐射场的快速发展，其也在飞速的占领着三维几何学习的重要位置。目前，该种表示已经可以在单个场景上实现非常快速且逼真的新视角合成和三维重建。至于其发展，首先未来的研究者可以做到更大更开放的场景空间，除此以外还可以应用在更有挑战性的光照环境，再就是去提升其泛化能力。
- 对隐式表示来说，连续隐式表示已成主流，并且已经有一些工作关注连续隐式表示的细节表达能力以及泛化性。和连续的点云表示一样，如何在保证泛化能力的前提下实现更好的高频特征和尖锐特征等，也会是未来持续的研究突破口。

### 6.2.2 多物体间的几何形变表示未来展望

在第四章和第五章当中，本文用了两种不同的表示来解决非刚性配准任务。第四章中的工作使得算法可以用一个神经网络回归非刚性配准，在第五章当中本文又进一步地将结构化信息输送到了网络当中使得神经网络有能力得到更鲁棒的配准结果。但是，目前为止也还有一些遗留的问题。

- 首先一点是结构化信息的自动提取。假如算法可以从待配准的模型当中自动提取结构化信息，而非采用额外输入节点图的方式，那么整体的流程将会有进一步的提升。未来工作可以将非刚性配准任务和自动创建模型结构化信息两者结合起来，从而实现更方便且鲁棒的多物体间的形变方案。
- 再一点就是处理源模型和目标模型之间的拓扑变化问题。在本文以上的研究中，总是认为源模型和目标模型之间是存在一个连续映射可以将其配准成功的，然而这在实际情况当中并不总是成立，例如从手臂收紧到手臂张开的过程。未来的工作可以将这种拓扑变化纳入考虑。
- 还有一点就是当源模型和目标模型之间的重叠度较小的情况下，比如对运动物体进行连续的捕捉，得到其不同运动状态和不同视角下的扫描结果。此时待配准的两个曲面之间并不存在完整的映射，对于两个曲面来说可配准区域都是局部而非整体。在这种情况下如何精确地识别可配准区域且进行合理的非刚性配准是非常具有挑战性的，也会是未来研究多物体间几何形变表示的重要难题。

## 参 考 文 献

- [1] Artillery sidewinder[EB/OL]. 2020. <https://artillery3d.com/>.
- [2] QIUYU W, JUYONG Z. A deep learning face recognition algorithm based on 3d alignment [J]. Journal of Systems Science and Mathematical Sciences, 2021.
- [3] WANG Y, LIU J, TANG X. Robust 3d face recognition by local shape difference boosting [J]. IEEE Trans. Pattern Anal. Mach. Intell., 2010.
- [4] CHEN X, MA H, WAN J, et al. Multi-view 3d object detection network for autonomous driving[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
- [5] WANG Y, CHAO W, GARG D, et al. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
- [6] LANG A H, VORA S, CAESAR H, et al. Pointpillars: Fast encoders for object detection from point clouds[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
- [7] Azure kinect dk[EB/OL]. 2019. <https://www.microsoft.com/en-us/p/azure-kinect-dk/8pp5vxmd9nhq?activetab=pivot:overviewtab>.
- [8] Einscan-pro+[EB/OL]. 2017. <https://www.einscan.com/handheld-3d-scanner/einscan-pro-plus>.
- [9] LUO S, HU W. Score-based point cloud denoising[C]//IEEE/CVF International Conference on Computer Vision (ICCV). 2021.
- [10] WANG Y, WU S, HUANG H, et al. Patch-based progressive 3 point set upsampling[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
- [11] YU L, LI X, FU C, et al. Pu-net: Point cloud upsampling network[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
- [12] ZHANG J, YAO Y, DENG B. Fast and robust iterative closest point[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2021.
- [13] JUBRAN I, MAALOUF A, KIMMEL R, et al. Provably approximated point cloud registration[C]//IEEE International Conference on Computer Vision (ICCV). 2021.
- [14] TRIEBEL R, KERSTING K, BURGARD W. Robust 3d scan point classification using associative markov networks[C]//Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006. 2006.
- [15] NGUYEN A, LE B. 3d point cloud segmentation: A survey[C]//2013 6th IEEE conference on robotics, automation and mechatronics (RAM). 2013.

- [16] KAZHDAN M M, HOPPE H. Screened poisson surface reconstruction[J]. ACM Transactions on Graphics (TOG), 2013.
- [17] LIAO Y, DONNÉ S, GEIGER A. Deep marching cubes: Learning explicit surface representations[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
- [18] QI C R, SU H, MO K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
- [19] WANG Y, SUN Y, LIU Z, et al. Dynamic graph CNN for learning on point clouds[J]. ACM Transactions on Graphics (TOG), 2019.
- [20] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Annual Conference on Neural Information Processing Systems. 2017.
- [21] ZHAO H, JIANG L, JIA J, et al. Point transformer[C]//IEEE International Conference on Computer Vision (ICCV). 2021.
- [22] DYN N, LEVINE D, GREGORY J A. A butterfly subdivision scheme for surface interpolation with tension control[J]. ACM Trans. Graph., 1990.
- [23] KOBBELT L. 3-subdivision[C]//Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 2000.
- [24] GARLAND M, HECKBERT P S. Surface simplification using quadric error metrics[C]// Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 1997.
- [25] Opensubdiv[EB/OL]. 2018. <http://graphics.pixar.com/opensubdiv>.
- [26] ZHANG J, DENG B, HONG Y, et al. Static/dynamic filtering for mesh geometry[J]. IEEE Trans. Vis. Comput. Graph., 2019.
- [27] WANG P, FU X, LIU Y, et al. Rolling guidance normal filter for geometric processing[J]. ACM Trans. Graph., 2015.
- [28] LIU H D, KIM V G, CHAUDHURI S, et al. Neural subdivision[J]. CoRR, 2020.
- [29] RAKOTOSAONA M, AIGERMAN N, MITRA N J, et al. Differentiable surface triangulation [J]. ACM Trans. Graph., 2021.
- [30] GUO Y, CAI L, ZHANG J. 3d face from X: learning face shape from diverse sources[J]. IEEE Trans. Image Process., 2021.
- [31] YU T, ZHENG Z, GUO K, et al. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
- [32] ALLEN B, CURLESS B, POPOVIC Z. The space of human body shapes: reconstruction and parameterization from range scans[J]. ACM Trans. Graph., 2003.
- [33] NEWCOMBE R A, IZADI S, HILLIGES O, et al. Kinectfusion: Real-time dense surface

- mapping and tracking[C]//IEEE International Symposium on Mixed and Augmented Reality (ISMAR). 2011.
- [34] NEWCOMBE R A, FOX D, SEITZ S M. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015.
- [35] TRAN L, LIU X. Nonlinear 3d face morphable model[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
- [36] CAO C, WENG Y, ZHOU S, et al. Facewarehouse: A 3d facial expression database for visual computing[J]. IEEE Trans. Vis. Comput. Graph. (TVCG), 2014.
- [37] LORENSEN W E, CLINE H E. Marching cubes: A high resolution 3d surface construction algorithm[C]//Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 1987.
- [38] TAN Q, GAO L, LAI Y, et al. Variational autoencoders for deforming 3d mesh models[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
- [39] CHEN Z, LI L, BRUNA J. Supervised community detection with line graph neural networks [C]//International Conference on Learning Representations (ICLR). 2019.
- [40] KHALIL E B, DAI H, ZHANG Y, et al. Learning combinatorial optimization algorithms over graphs[C]//Annual Conference on Neural Information Processing Systems (NIPS). 2017.
- [41] SCARSELLI F, GORI M, TSOI A C, et al. The graph neural network model[J]. IEEE Trans. Neural Networks, 2009.
- [42] BRUNA J, ZAREMBA W, SZLAM A, et al. Spectral networks and locally connected networks on graphs[C]//International Conference on Learning Representations (ICLR). 2014.
- [43] DEFFERRARD M, BRESSON X, VANDERGHEYNST P. Convolutional neural networks on graphs with fast localized spectral filtering[C]//Annual Conference on Neural Information Processing Systems (NIPS). 2016.
- [44] EITEL A, SPRINGENBERG J T, SPINELLO L, et al. Multimodal deep learning for robust RGB-D object recognition[C]//International Conference on Intelligent Robots and Systems (IROS). 2015.
- [45] BOZIC A, PALAFOX P R, ZOLLMÖFER M, et al. Neural non-rigid tracking[C]//Annual Conference on Neural Information Processing Systems (NeurIPS). 2020.
- [46] FENG J, WANG Y, CHANG S. 3d shape retrieval using a single depth image from low-cost sensors[C]//Winter Conference on Applications of Computer Vision (WACV). 2016.
- [47] SCHWARZ M, SCHULZ H, BEHNKE S. RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features[C]//International Conference on Robotics and Automation (ICRA). 2015.

- 
- [48] ZHAO J, XIE X, XU X, et al. Multi-view learning overview: Recent progress and new challenges[J]. Inf. Fusion, 2017.
  - [49] SU H, MAJI S, KALOGERAKIS E, et al. Multi-view convolutional neural networks for 3d shape recognition[C]//IEEE International Conference on Computer Vision (ICCV). 2015.
  - [50] ZANUTTIGH P, MINTO L. Deep learning for 3d shape classification from multiple depth maps[C]//IEEE International Conference on Image Processing (ICIP). 2017.
  - [51] WANG C, PELILLO M, SIDDIQI K. Dominant set clustering and pooling for multi-view 3d object recognition[C]//British Machine Vision Conference (BMVC). 2017.
  - [52] KANEZAKI A, MATSUSHITA Y, NISHIDA Y. Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
  - [53] SITZMANN V, ZOLLMÖFER M, WETZSTEIN G. Scene representation networks: Continuous 3d-structure-aware neural scene representations[C]//Annual Conference on Neural Information Processing Systems (NeurIPS). 2019.
  - [54] MILDENHALL B, SRINIVASAN P P, TANCIK M, et al. Nerf: Representing scenes as neural radiance fields for view synthesis[C]//European Conference on Computer Vision (ECCV). 2020.
  - [55] RIEGLER G, ULUSOY A O, BISCHOF H, et al. Octnetfusion: Learning depth fusion from data[C]//International Conference on 3D Vision (3DV). 2017.
  - [56] RIEGLER G, OSMAN ULUSOY A, GEIGER A. Octnet: Learning deep 3d representations at high resolutions[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
  - [57] PARK J J, FLORENCE P, STRAUB J, et al. Deepsdf: Learning continuous signed distance functions for shape representation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
  - [58] GENOVA K, COLE F, VLASIC D, et al. Learning shape templates with structured implicit functions[C]//IEEE International Conference on Computer Vision (ICCV). 2019.
  - [59] MESCHEDER L M, OECHSLE M, NIEMEYER M, et al. Occupancy networks: Learning 3d reconstruction in function space[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
  - [60] CHEN Z, ZHANG H. Learning implicit fields for generative shape modeling[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
  - [61] NIEMEYER M, MESCHEDER L, OECHSLE M, et al. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

- 
- [62] YARIV L, KASTEN Y, MORAN D, et al. Multiview neural surface reconstruction by disentangling geometry and appearance[C]//Annual Conference on Neural Information Processing Systems (NeurIPS). 2020.
  - [63] CHIBANE J, ALLDIECK T, PONS-MOLL G. Implicit functions in feature space for 3d shape reconstruction and completion[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
  - [64] YAO S, YANG F, CHENG Y, et al. 3d shapes local geometry codes learning with sdf[C]// IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). 2021.
  - [65] TRETSCHK E, TEWARI A, GOLYANIK V, et al. Patchnets: Patch-based generalizable deep implicit 3d shape representations[C]//Proc. IEEE/CVF Int. Conf. Comput. Vis. 2020.
  - [66] JIANG C M, SUD A, MAKADIA A, et al. Local implicit grid representations for 3d scenes [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
  - [67] BESL P J, MCKAY N D. A method for registration of 3-d shapes[J]. IEEE Trans. Pattern Anal. Mach. Intell., 1992.
  - [68] BOUAZIZ S, TAGLIASACCHI A, PAULY M. Sparse iterative closest point[J]. Comput. Graph. Forum, 2013.
  - [69] RUSINKIEWICZ S, LEVOY M. Efficient variants of the ICP algorithm[C]//International Conference on 3D Digital Imaging and Modeling (3DIM). 2001.
  - [70] SEGAL A, HÄHNEL D, THRUN S. Generalized-icp[C]//Robotics: Science and Systems. 2009.
  - [71] SERVOS J, WASLANDER S L. Multi channel generalized-icp[C]//IEEE International Conference on Robotics and Automation (ICRA). 2014.
  - [72] FISCHLER M A, BOLLES R C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography[J]. Commun. ACM, 1981.
  - [73] RUSU R B, BLODOW N, BEETZ M. Fast point feature histograms (FPFH) for 3d registration[C]//IEEE International Conference on Robotics and Automation (ICRA). 2009.
  - [74] ROSEN D M, CARLONE L, BANDEIRA A S, et al. Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group[J]. Int. J. Robotics Res., 2019.
  - [75] IZATT G, DAI H, TEDRAKE R. Globally optimal object pose estimation in point clouds with mixed-integer programming[C]//International Symposium on Robotics Research (ISRR). 2017.
  - [76] YANG J, LI H, CAMPBELL D, et al. Go-icp: A globally optimal solution to 3d ICP point-set registration[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2016.
  - [77] YEW Z J, LEE G H. Rpm-net: Robust point matching using learned features[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

- 
- [78] ZENG A, SONG S, NIESSNER M, et al. 3dmatch: Learning local geometric descriptors from RGB-D reconstructions[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
  - [79] YEW Z J, LEE G H. 3dfeat-net: Weakly supervised local 3d features for point cloud registration[C]//European Conference on Computer Vision (ECCV). 2018.
  - [80] AOKI Y, GOFORTH H, SRIVATSAN R A, et al. Pointnetlk: Robust & efficient point cloud registration using pointnet[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
  - [81] SARODE V, LI X, GOFORTH H, et al. Pcrnet: Point cloud registration network using pointnet encoding[J]. CoRR, 2019.
  - [82] WANG Y, SOLOMON J. Deep closest point: Learning representations for point cloud registration[C]//IEEE International Conference on Computer Vision (ICCV). 2019.
  - [83] WANG Y, SOLOMON J M. Prnet: Self-supervised learning for partial-to-partial registration [C]//Annual Conference on Neural Information Processing Systems (NIPS). 2019.
  - [84] HUANG X, MEI G, ZHANG J. Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
  - [85] MYRONENKO A, SONG X B. Point set registration: Coherent point drift[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2010.
  - [86] KOLESOV I, LEE J, SHARP G, et al. A stochastic approach to diffeomorphic point set registration with landmark constraints[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2016.
  - [87] MA J, ZHAO J, YUILLE A L. Non-rigid point set registration by preserving global and local structures[J]. IEEE Trans. Image Process., 2016.
  - [88] GOLYANIK V, TAETZ B, REIS G, et al. Extended coherent point drift algorithm with correspondence priors and optimal subsampling[C]//Winter Conference on Applications of Computer Vision (WACV). 2016.
  - [89] HIROSE O. A bayesian formulation of coherent point drift[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2020.
  - [90] WANG L, FANG Y. Coherent point drift networks: Unsupervised learning of non-rigid point set registration[J]. CoRR, 2019.
  - [91] LIU X, QI C R, GUIBAS L J. Flownet3d: Learning scene flow in 3d point clouds[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
  - [92] KITTENPLON Y, ELDAR Y C, RAVIV D. Flowstep3d: Model unrolling for self-supervised scene flow estimation[J]. CoRR, 2020.
  - [93] MITTAL H, OKORN B, HELD D. Just go with the flow: Self-supervised scene flow es-

- timation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
- [94] CHUI H, RANGARAJAN A. A new point matching algorithm for non-rigid registration[J]. Comput. Vis. Image Underst., 2003.
- [95] JIAN B, VEMURI B C. Robust point set registration using gaussian mixture models[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2011.
- [96] CHEN J, MA J, YANG C, et al. Non-rigid point set registration via coherent spatial mapping [J]. Signal Process., 2015.
- [97] YANG Y, ONG S H, FOONG K W C. A robust global and local mixture distance based non-rigid point set registration[J]. Pattern Recognit., 2015.
- [98] WANG L, CHEN J, LI X, et al. Non-rigid point set registration networks[J]. CoRR, 2019.
- [99] SUMNER R W, SCHMID J, PAULY M. Embedded deformation for shape manipulation[J]. ACM Trans. Graph., 2007.
- [100] YAO Y, DENG B, XU W, et al. Quasi-newton solver for robust non-rigid registration[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
- [101] LI H, SUMNER R W, PAULY M. Global correspondence optimization for non-rigid registration of depth scans[J]. Comput. Graph. Forum, 2008.
- [102] HOPPE H, DEROSE T, DUCHAMP T, et al. Surface reconstruction from unorganized points [C]//Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 1992.
- [103] HELD R, GUPTA A, CURLESS B, et al. 3d puppetry: a kinect-based interface for 3d animation[C]//ACM Symposium on User Interface Software and Technology (UIST). 2012.
- [104] SANTANA J M, WENDEL J, TRUJILLO A, et al. Multimodal location based services— semantic 3d city data as virtual and augmented reality[M]//Progress in location-based services (LBS). 2017.
- [105] LI R, LI X, FU C, et al. PU-GAN: A point cloud upsampling adversarial network[C]// IEEE/CVF International Conference on Computer Vision (ICCV). 2019.
- [106] LI R, LI X, HENG P, et al. Point cloud upsampling via disentangled refinement[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021.
- [107] QIAN Y, HOU J, KWONG S, et al. Pugeo-net: A geometry-centric network for 3d point cloud upsampling[C]//European Conference on Computer Vision (ECCV). 2020.
- [108] QIAN G, ABUALSHOUR A, LI G, et al. PU-GCN: point cloud upsampling using graph convolutional networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021.
- [109] Sketchfab[EB/OL]. <https://sketchfab.com>.

- 
- [110] CORSINI M, CIGNONI P, SCOPIGNO R. Efficient and flexible sampling with blue noise properties of triangular meshes[J]. IEEE Transactions on Visualization and Computer Graphics (TVCG), 2012.
  - [111] PASZKE A, GROSS S, MASSA F, et al. Pytorch: An imperative style, high-performance deep learning library[C]//Annual Conference on Neural Information Processing Systems (NeurIPS). 2019.
  - [112] XU D, DUAN Q, ZHENG J, et al. Recovering surface details under general unknown illumination using shading and coarse multi-view stereo[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2014.
  - [113] DAI A, NIESSNER M, ZOLLHÖFER M, et al. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration[J]. ACM Trans. Graph., 2017.
  - [114] SHIMADA S, GOLYANIK V, TRETSCHK E, et al. Dispvoxnets: Non-rigid point set alignment with supervised learning proxies[C]//International Conference on 3D Vision. 2019.
  - [115] ZHANG C, PUJADES S, BLACK M J, et al. Detailed, accurate, human shape estimation from clothed 3d scan sequences[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
  - [116] CHOI S, ZHOU Q, MILLER S, et al. A large dataset of object scans[A]. 2016.
  - [117] KIL Y J, MEDEROS B, AMENTA N. Laser scanner super-resolution[C]//Symposium on Point Based Graphics. 2006.
  - [118] HAMDI-CHERIF A, DIGNE J, CHAINE R. Super-resolution of point set surfaces using local similarities[J]. Comput. Graph. Forum, 2018.
  - [119] YU L, LI X, FU C, et al. Ec-net: An edge-aware point set consolidation network[C]// European Conference on Computer Vision (ECCV). 2018.
  - [120] CATMULL E, CLARK J. Recursively generated b-spline surfaces on arbitrary topological meshes[J]. Computer-aided design, 1978.
  - [121] LOOP C. Smooth subdivision surfaces based on triangles[J]. Master's thesis, University of Utah, Department of Mathematics, 1987.
  - [122] HOPPE H, DEROSE T, DUCHAMP T, et al. Piecewise smooth surface reconstruction[C]// Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 1994.
  - [123] TAUBIN G. Introduction to geometric processing through optimization[J]. IEEE Computer Graphics and Applications, 2012.
  - [124] VALLET B, LÉVY B. Spectral geometry processing with manifold harmonics[J]. Comput. Graph. Forum, 2008.
  - [125] ARVANITIS G, LALOS A S, MOUSTAKAS K, et al. Feature preserving mesh denoising based on graph spectral processing[J]. IEEE Trans. Vis. Comput. Graph., 2019.

- 
- [126] DA FONA COSTA L, CESAR JR R M. Shape classification and analysis: theory and practice [M]. 2018.
  - [127] LUCIANO L, HAMZA A B. Deep learning with geodesic moments for 3d shape classification[J]. Pattern Recognition Letters, 2018.
  - [128] PARK J J, FLORENCE P, STRAUB J, et al. Deepsdf: Learning continuous signed distance functions for shape representation[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
  - [129] CARR J C, BEATSON R K, CHERRIE J B, et al. Reconstruction and representation of 3d objects with radial basis functions[C]//Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 2001.
  - [130] YING L, ZORIN D. A simple manifold-based construction of surfaces of arbitrary smoothness[J]. ACM Trans. Graph., 2004.
  - [131] GU X, HE Y, QIN H. Manifold splines[C]//ACM Symposium on Solid and Physical Modeling (SPM). 2005.
  - [132] VECCHIA G D, JÜTTLER B, KIM M. A construction of rational manifold surfaces of arbitrary topology and smoothness from triangular meshes[J]. Comput. Aided Geom. Des., 2008.
  - [133] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
  - [134] NIESSNER M, LOOP C T, MEYER M, et al. Feature-adaptive GPU rendering of catmull-clark subdivision surfaces[J]. ACM Trans. Graph., 2012.
  - [135] LIU H D, KIM V G, CHAUDHURI S, et al. Neural subdivision[J]. ACM Trans. Graph., 2020.
  - [136] KAMMERL J. pcl[EB/OL]. 2013. <http://www.pointclouds.org/>.
  - [137] KNAPITSCH A, PARK J, ZHOU Q Y, et al. Tanks and temples: Benchmarking large-scale scene reconstruction[J]. ACM Trans. Graph., 2017.
  - [138] QIAN C, SUN X, WEI Y, et al. Realtime and robust hand tracking from depth[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2014.
  - [139] VEDULA S, BAKER S, RANDER P, et al. Three-dimensional scene flow[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2005.
  - [140] ZHAO S, DONG Y, CHANG E I, et al. Recursive cascaded networks for unsupervised medical image registration[C]//IEEE International Conference on Computer Vision (ICCV). 2019.
  - [141] SENTKER T, MADESTA F, WERNER R. GDL-FIRE4d : Deep learning-based fast 4d CT image registration[C]//Medical Image Computing and Computer Assisted Intervention

- (MICCAI). 2018.
- [142] LU W, WAN G, ZHOU Y, et al. Deepicp: An end-to-end deep neural network for 3d point cloud registration[J]. CoRR, 2019.
- [143] CHOY C, DONG W, KOLTUN V. Deep global registration[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
- [144] TOMBARI F, SALTÌ S, DI STEFANO L. Unique signatures of histograms for local surface description[C]//DANIILIDIS K, MARAGOS P, PARAGIOS N. European Conference on Computer Vision (ECCV). 2010.
- [145] BOOKSTEIN F L. Principal warps: Thin-plate splines and the decomposition of deformations[J]. IEEE Trans. Pattern Anal. Mach. Intell., 1989.
- [146] AMBERG B, ROMDHANI S, VETTER T. Optimal step nonrigid ICP algorithms for surface registration[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2007.
- [147] WU Z, SONG S, KHOSLA A, et al. 3d shapenets: A deep representation for volumetric shapes[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015.
- [148] CHO K, VAN MERRIENBOER B, BAHdanaud D, et al. On the properties of neural machine translation: Encoder-decoder approaches[C]//Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. 2014.
- [149] TEED Z, DENG J. RAFT: recurrent all-pairs field transforms for optical flow[C]//Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part II. 2020.
- [150] CHEN D, TIAN X, SHEN Y, et al. On visual similarity based 3d model retrieval[J]. Comput. Graph. Forum, 2003.
- [151] GAO L, YANG J, QIAO Y, et al. Automatic unpaired shape deformation transfer[J]. ACM Trans. Graph., 2018.
- [152] LIU S, CHEN W, LI T, et al. Soft rasterizer: Differentiable rendering for unsupervised single-view mesh reconstruction[J]. CoRR, 2019.
- [153] TAN F, ZHU H, CUI Z, et al. Self-supervised human depth estimation from monocular videos [C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
- [154] VLASIC D, BARAN I, MATUSIK W, et al. Articulated mesh animation from multi-view silhouettes[J]. ACM Trans. Graph., 2008.
- [155] VAROL G, ROMERO J, MARTIN X, et al. Learning from synthetic humans[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.
- [156] LOPER M, MAHMOOD N, ROMERO J, et al. SMPL: a skinned multi-person linear model [J]. ACM Trans. Graph., 2015.

- [157] BRONSTEIN A M, BRONSTEIN M M, KIMMEL R. Numerical geometry of non-rigid shapes[M]. 2009.
- [158] GAO L, LAI Y, YANG J, et al. Sparse data driven mesh deformation[J]. CoRR, 2017.
- [159] BOGO F, ROMERO J, PONS-MOLL G, et al. Dynamic FAUST: Registering human bodies in motion[C]//CVPR. 2017.
- [160] GROUEIX T, FISHER M, KIM V G, et al. 3d-coded: 3d correspondences by deep deformation[C]//European Conference on Computer Vision (ECCV). 2018.
- [161] ZHOU Q, PARK J, KOLTUN V. Fast global registration[C]//European Conference on Computer Vision (ECCV). 2016.
- [162] RUSTAMOV R M. Laplace-beltrami eigenfunctions for deformation invariant shape representation[C]//Eurographics Symposium on Geometry Processing. 2007.
- [163] SUN J, OVSJANIKOV M, GUIBAS L J. A concise and provably informative multi-scale signature based on heat diffusion[J]. Comput. Graph. Forum, 2009.
- [164] HALIMI O, LITANY O, RODOLÀ E, et al. Unsupervised learning of dense shape correspondence[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
- [165] AYGÜN M, LÄHNER Z, CREMERS D. Unsupervised dense shape correspondence using heat kernels[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
- [166] ZENG Y, QIAN Y, ZHU Z, et al. Cornet3d: Unsupervised end-to-end learning of dense correspondence for 3d point clouds[J]. CoRR, 2020.
- [167] FENG W, ZHANG J, CAI H, et al. Recurrent multi-view alignment network for unsupervised surface registration[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021.
- [168] GUO K, XU F, WANG Y, et al. Robust non-rigid motion tracking and surface reconstruction using L0 regularization[C]//IEEE International Conference on Computer Vision (ICCV). 2015.
- [169] LI H, ADAMS B, GUIBAS L J, et al. Robust single-view geometry and motion reconstruction [J]. ACM Trans. Graph., 2009.
- [170] TEED Z, DENG J. RAFT-3D: scene flow using rigid-motion embeddings[J]. CoRR, 2020.
- [171] SMITH L N, TOPIN N. Super-convergence: Very fast training of residual networks using large learning rates[J]. CoRR, 2017.
- [172] KINGMA D P, BA J. Adam: A method for stochastic optimization[C]//International Conference on Learning Representations (ICLR). 2015.
- [173] LI C, SIMON T, SARAGIH J M, et al. LBS autoencoder: Self-supervised fitting of artic-

- ulated meshes to point clouds[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
- [174] LITANY O, REMEZ T, RODOLÀ E, et al. Deep functional maps: Structured prediction for dense shape correspondence[C]//IEEE International Conference on Computer Vision (ICCV). 2017.

## 致 谢

这是一篇修改了很多次的致谢。反复斟酌，只因在中国科大的这十年，注定会是常常被回忆起的十年。时至今日，这一段求学的过程并没有想象中曲折，这让我感觉自己十分幸运。这期间需要感谢的人和事太多，全部提到我只怕篇幅不够，故在此列举最重要的若干致谢对象。

首先最感谢的是我的导师张举勇老师。张老师在学术上很有自己的想法和追求，也非常注重对学生科研品味和技能的培养，并用自己积极且负责的工作态度感染着大家不断努力。在最开始我学术能力非常差的时候，老师给予了我充分的耐心、引导，最重要的是给了我试错的底气，才使得我有机会走向正确的方向。在科研习惯方面，老师也给我们树立了很好的榜样，使得我们在离开学校之际能够尽可能做一个完善的科研人，支撑我们后续的发展。除此以外也（不分先后次序地）感谢同组的刘利刚老师，山东大学的周元峰、辛士庆老师，英国卡迪夫大学的邓柏林老师，香港城市大学的侯军辉老师，桂林电子科大的罗笑南老师，浙江大学的鲍虎军老师对我的科研课题、论文发表的帮助和指导。

其次我要感谢我的母校，中国科学技术大学。感谢学校给我们提供了完善的学习、科研环境，让我们能安心搞科研，不受外界嘈杂的干扰。还要感谢学校对学生的关怀、耐心甚至无限的包容。用一句歌词来形容，那应该是“你的怀抱是温暖的海洋”。在这十年里，有很多瞬间让我感觉到了“母校”二字的沉甸。这十年是我成长巨大的十年，临别之际，只觉得母校的每一寸土地都那么让人留恋。踏出母校大门之后，我希望自己能带着“科大人”的深刻烙印不断在科学的道路上探索、进取，努力把握所有能给母校带来荣誉的机会。同时，我也不会忘记母校名字中的“科学技术”四个字，把眼光放长远，不管身在何处、什么岗位，心中始终会默记学校给我们教导的初心，尽可能为整个人类社会的科学技术发展作出自己微薄的贡献。

感谢所有在科研中帮助过我的小伙伴：尤其（但不仅限于）感谢玉东、博艺、豪飞、宇馨解决过我关键的问题，以及泓锐在我的多篇一作论文中均提供了不可或缺的帮助，当然还有其余所有的 3DV 组的同学们与我一起度过这段难忘的科研生活。

感谢我的女朋友凡凡一直支持我的选择，热爱我的事业，陪伴我的生活，缓解我的焦虑。感谢曾经炫音社的小伙伴们给我带来了很多的快乐，感谢叶晓雯老师在音乐上对我的深刻教导。感谢本科和研究生阶段所有的室友。感谢我的家人，尤其是父母，一直给予的经济和精神上的双重支持。感谢睿睿陪我从高中到研究生一起度过了共同的八年。感谢我最重要的朋友小雨。

## 致 谢

---

此外，作为一个重度音乐依赖者，请容许我占用一些小小的篇幅来感谢我喜爱的音乐艺术家们。首先是我最尊敬的词曲作家、吉他手水野良树先生和山下穗尊先生。还有我个人最喜欢的歌手吉冈圣惠。除此以外，也感谢其他经常陪在我耳朵里面的歌手，如周杰伦，蔡恩雨，邓紫棋，冯提莫等。

感谢这十年时间所带给我的一切。我会先把这些收拾好，不会过分追忆，因为明白还有很长的职业生涯的路要走；但是偶尔想起来可能会感慨万千，因为有太多值得反思的成长，都会成为后续人生道路中的无形财富。

## 在读期间发表的学术论文与取得的研究成果

### 已发表论文

1. Wanquan Feng, Jin Li, Hongrui Cai, Xiaonan Luo, and Juyong Zhang. Neural Points: Point Cloud Representation with Neural Fields for Arbitrary Upsampling. *In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
2. Wanquan Feng, Juyong Zhang, Hongrui Cai, Haofei Xu, Junhui Hou, and Hujun Bao. Recurrent Multi-View Alignment Network for Unsupervised Surface Registration. *In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
3. Wanquan Feng, Juyong Zhang, Yuanfeng Zhou, and Shiqing Xin. GDR-Net: A Geometric Detail Recovering Network for 3D Scanned Objects. *In IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 2021.
4. Zhuang Peng, Boyi Jiang, Haofei Xu, Wanquan Feng and Juyong Zhang. Facial Optical Flow Estimation via Neural Non-rigid Registration. *In Computational Visual Media (CVMJ)*, 2022.

### 待发表论文

1. Wanquan Feng, Hongrui Cai, Junhui Hou, Bailin Deng and Juyong Zhang. Differentiable Deformation Graph based Neural Non-rigid Registration. *Submitted to Communications in Mathematics and Statistics (CIMS)*, 2022.
2. Zhongqi Yang, Zhuang Peng, Wanquan Feng, Yang Hong and Juyong Zhang. TUDF-NeRF: Generalizable Neural Radiance Field via Truncated Unsigned Distance Field. *Submitted to European Conference on Computer Vision (ECCV)*, 2022.
3. Hongrui Cai, Wanquan Feng and Juyong Zhang. Neural Surface Reconstruction of Dynamic Scenes with Monocular RGB-D Camera. *Submitted to Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2022.