分类号 TP391.9 学校代码 10495
UDC 681.5 密 级 公开

武汉纺织大学

硕士学位论文

隐式神经表面重建的改进研究

作者姓名： 于楚飞
学 号 ： 2015373044
指导教师： 苏工兵
学科门类： 工学
专 业： 机械
研究方向： 3D 视觉
完成日期： 二零二三年六月

Wuhan Textile *University*

M. E. Dissertation

Improved study of implicit neural surface reconstruction

Candidate： **Yu Chufei**

Supervisor： **Su Gongbing**

Time: **June 2023**

摘　要

表面重建是指通过 3D 采集设备获取的点云或图像推断出物体、场景等 3D 对象表面的过程。常见的表面表示方法一般为体素或网格。体素表示支持与图像处理兼容的卷积操作，但会带来立方级别增长的内存开销，因此难以获取平滑和精细的几何表面。网格表示一般通过微调预定义模板的网络的映射参数来拟合表面，但其对网格划分和拓扑结构非常敏感，应用场景受限。

近几年，出现了一种可以利用神经网络对物体、场景等 3D 对象表面进行隐式建模的方法。它将几何表面的结构信息编码为神经网络的参数，无需获取显式的表面位置信息。此类方法可以处理任意拓扑结构的对象，并且兼容图像和点云两种不同的输入形式。尽管该方法具有很大的潜力，但目前其重建精度和鲁棒性方面还有不足，并且存在效率低和拓展性差等问题。为此，开展了以现有图像和点云两种不同的输入形式的隐式神经表面重建为基础的改进研究，以期将其扩展到真实世界的重建任务中。

基于点云的隐式神经表面重建通常采用自动解码器的架构，它将表面表示为隐式函数的零等值面，并利用表面附近带监督值的 3D 点对表面拟合。为了提高重建质量，以 Deepsdf 算法框架为基础进行了改进。具体而言，设计了一种能够提取邻域表面特征的编码器，从而增强了其对复杂局部细节的恢复能力。此外，使用了一种自适应的损失加权策略，增加了采样点的利用率。对 ShapeNet、GSO 数据集测试的结果表明，相比改进前，重叠度和 F-score 分别提高了 1.458%和 1.46%，平均倒角距离降低了 0.08，证明了改进点的有效性。

基于图像的隐式神经表面重建一般采用神经辐射场架构，通过神经渲染图像序列或单张图像来重建表面。为了提高重建质量，以 Neus 算法框架为基础进行了改进。具体而言，提出了一种带静态注意力机制的网络架构，引入了新的归纳偏执，从而增强了网络的精确参数化和对抗高频噪声的能力。此外，使用球谐函数对非朗伯场景的光场进行显式建模，降低了模型的拟合难度；设计了一种体素缓存结构来记录场景空间，并指导网络对表面附近进行针对性训练，从而增加了训练效率。对 DTU 数据集测试的结果表明，在使用掩码和不使用掩码的情况下，相比改进前，平均倒角距离分别降低了 0.1 和 0.06，证明了改进点的有效性。

上述以图像和点云两种不同的输入形式隐式表面重建的改进均能显著提高相应评价指标的成绩，并提供了更好的视觉效果，同时具有更好的适应性和普适性。为实现更高精度和更可靠的表示重建提供了重要的技术支持。

关键词：表面重建；神经辐射场；符号距离函数；多层感知机；注意力机制
研究类型：应用研究

Abstract

Surface reconstruction refers to the process of inferring the surface of a 3D object, scene, etc. from a point cloud or image acquired by a 3D capture device. The commonly used surface representation methods are voxel and mesh. Voxel representation supports convolution operations compatible with image processing but incurs memory overhead that grows cubically, making it difficult to obtain smooth and fine geometric surfaces. Mesh representation is generally achieved by adjusting the mapping parameters of a pre-defined template network to fit the surface, but it is sensitive to mesh partitioning and topology, limiting its application scenarios.

In recent years, a method has emerged that can use neural networks to implicitly model the surface of 3D objects, scenes, etc. It encodes the geometric structure of the surface as the parameters of a neural network without obtaining explicit surface position information. Such methods can handle objects of arbitrary topological structures and is applicable to two different input forms, images and point clouds. Although this method has great potential, there are still shortcomings in terms of reconstruction accuracy and robustness, as well as issues with low efficiency and poor scalability. To address these, improvement research based on implicit neural surface reconstruction using the existing two different input forms of images and point clouds has been conducted, with the aim of extending it to real-world reconstruction tasks.

Implicit neural surface reconstruction based on point clouds typically uses an auto-encoder architecture, which represents the surface as the zero-level set of an implicit function and fits the surface using 3D points with supervised values near the surface. To improve the quality of reconstruction, improvements have been made based on the Deepsdf algorithm framework. Specifically, an encoder capable of extracting neighborhood surface features was designed to enhance its recovery ability for complex local details. In addition, an adaptive loss weighting strategy was used to increase the utilization of sampling points. Testing on the ShapeNet and GSO datasets showed that compared to before the improvement, the overlap ratio and F-score increased by 1.458% and 1.46%, respectively, and the average chamfer distance decreased by 0.08, demonstrating the effectiveness of the improvements.

Implicit neural surface reconstruction based on images generally uses a neural radiance field architecture to reconstruct the surface by rendering image sequences or single images

with neural networks. To improve the quality of reconstruction, improvements have been made based on the Neus algorithm framework. Specifically, a network architecture with a static attention mechanism was proposed, and new inductive biases were introduced to enhance the network's ability to accurately parameterize and counter high-frequency noise. In addition, spherical harmonics were used to explicitly model the non-Lambertian scene's light field, reducing the model's fitting difficulty. A voxel cache structure was designed to record the scene space and guide the network to perform targeted training near the surface, thereby increasing training efficiency. Testing on the DTU dataset showed that, with and without masks, compared to before the improvement, the average chamfer distance decreased by 0.1 and 0.06, respectively, demonstrating the effectiveness of the improvements.

Improvements in implicit surface reconstruction using two different input forms of images and point clouds have significantly improved the corresponding evaluation metrics and provided better visual effects, while also having better adaptability and universality. This provides important technical support for achieving higher precision and more reliable representation reconstruction.

Key words： surface reconstruction; neural radiation field; signed distance function**;** multi-layer perceptron；attention mechanism

Thesis： applied research

目　录

1 绪论

1.1 研究背景与意义

表面重建是三维重建领域中的一个关键的技术环节，其目的是从图像、点云或其他形式的非结构化的原始数据中恢复待重建对象表面的几何信息。表面重建的结果通常采用显式网格的形式。网格对 3D 对象的平滑表面进行了片状的线性近似，能够充分利用图形处理器的并行处理能力，使其易于编辑和可视化。随着元宇宙、虚拟现实（AR/VR）和人工智能（AI）等领域的快速发展，表面重建已成为这些领域的研究热点之一。

在表面重建领域的研究中，基于点云和基于图像的表面重建是两个最主要的方向。基于点云的表面重建一般以传统的计算几何学方法为基础，通过构造基于图的数据结构来建立点云与网格顶点之间的双射关系。虽然此类方法在最大程度地保持了重建网格与输入点云之间的几何特征一致性，但是其非常依赖输入数据的质量，极易受到部分噪声点云的干扰。基于图片的表面重建一般以传统的多视图几何方法为基础，它主要分为两个阶段：点云重构和表面恢复。点云重构是指通过图片的相机内外参数，将其所有像素点逆向投影到三维空间中，并重构其三维坐标的过程。而表面恢复指是通过分析像素与三维点的映射关系，推断出对象的三维形状和表面信息的过程。虽然此类方法能够从任意角度和距离的图片中获取 3D 对象的表面信息，但是需要对图像进行非常复杂的处理，同时对采集过程中的光照变化非常敏感。

近年来，隐式神经表示在表面重建领域取得了大量的研究进展。相比点云、网格和体素等传统的离散表示形式，隐式神经表示认为 3D 对象的轮廓面是一个连续隐式函数的零水平集，通过训练神经网络为任意的三维坐标分配一个对应的量化值，能够对具有任意拓扑结构的 3D 对象进行参数化，在节约计算资源和抗噪声干扰方面有巨大的优势。基于点云的隐式神经表面重建通过引入对象类别的形状先验，来弥补输入点云的孔洞和噪声对表面重建的负面影响。基于图片的隐式神经表面重建通过使用体积渲染直接从一组包含位姿的图像中学习 3D 对象的形状和纹理等几何信息，绕过了传统方法中点云重构这一中间步骤，降低了表面重建对内存的开销和对输入数据的分辨率限制。此外，隐式神经表示还是一种流形表示，对其使用等值面抽取技术获得是网格水密的，即网格中的每个三角面片的任意一边都有一个其他三角面的边与之相连。

尽管基于点云和图像的隐式神经表面重建研究分别取得了一定的进展，相比传统方法具有更高的鲁棒性和效率，但是此类方法在面对复杂的 3D 对象时，重建结果在精度和准确性方面仍存在不足，难以扩展到真实世界的表面重建任务当中。因此，隐式神经表面重建当前仍然有很大的研究空间，值得对其进行进一步的探索和改进。

1.2 国内外研究现状

与传统的离散场景表示不同，隐式神经表示通过训练一个深度神经网络来拟合一个描述了 3D 对象表面的连续隐式函数，它支持以任意空间分辨率进行采样并且在节约计算资源方面有着巨大的优势。神经网络参数化了任意空间坐标与该位置对应的量化值之间的映射关系,并且对 3D 对象的拓扑结构无任何限制。按照输入数据形式的不同，隐式神经表面重建主要可以分为基于点云和基于图像的方法。

1.2.1 基于点云的隐式神经表面重建

一些研究者尝试使用不同的建模方法对 3D 对象进行隐式表示。Mescheder[1]等人通过使用神经网络学习 3D 对象的二元决策边界来对其进行参数化，该边界划分了空间中所有被 3D 对象占用的区域。Park 等人[2]使用符号距离函数（signed distance function，SDF）来参数化 3D 物体，并且将其表面的几何参数分别解耦成表示形状先验的网络参数和表示个体特征的低维向量。Gropp 等人[3]使用了程函方程（Eikonal equation）[4]对 3D 对象的表面法向量进行正则化，能够只从表面点云中重构 3D 对象的表面，避免了在其表面的附近进行复杂的补充采样。Sitzmann 等人[5]使用基于梯度的元学习框架对隐式表示进行重新建模，实现了对推理过程的加速。

一些研究者尝试通过现有方法进行改进来提升隐式神经表示的在某些方面的性能。Duan Yueqi 等人[6]采用了由粗到细的阶段性学习课程来提升隐式神经表面重建方法的精确性和鲁棒性，该方法引导模型首先关注目标形状的全局轮廓，随后再将注意力转移到局部细节上。Martel 等人[7]认为现有方法在对具有不同细节和规模的复杂场景进行建模时存在局限性，因此使用自适应坐标网络以分层方式表示场景，该层次结构的每个级别都捕获不同级别的细节和比例。Tretschk 等人[8]将 3D 对象分解成独立的子块并分别进行参数化，随后再将其逆向组合成一个整体，降低了表示难度并提高了泛化能力。席建悦等人[9]使用了 B 样条[10]对 3D 对象的表面进行编码，并使用了长短期记忆网络（Long Short-Term Memory，LSTM）[11]来挖掘潜在空间中的几何结构信息。

一些研究者尝试将隐式神经表示扩展到了更多的实际应用场景中。Mu 等人[12]使用 SDF 来隐式表示关节形状的新方法，它能够从整体形状中解耦出铰接的角度，并且通过编辑对该角度来生成新的几何形状。Saito[13]等人通过将输入图片与人体的几何先验进行对齐，能够从单视角输入中快速的重建出高分辨的穿衣数字化人体。

1.2.2 基于图像的隐式神经表面重建

近几年，Mildenhall 等人[14]提出了神经辐射场（neural radiance fields，NeRF），它能够从一组包含姿态信息的图像中学习场景的隐式表示，并通过神经渲染技术，在任意的观察视角下，合成高分辨率的观测图像。NeRF 实质是通过执行可微的体积渲染和反向

传播合成图像的光度误差来重现 3D 场景，能够直接从 2D 图像中学习 3D 场景的几何信息。与先前的隐式神经表示不同，NeRF 降低对输入数据的严格要求，极大地扩展了其应用前景，并且愈发的受到计算机视觉和计算机图形图像学等领域的关注。然而，NeRF 将 3D 场景建模为一团密度和颜色分布不均匀的云雾状的介质，没有对 3D 对象的几何表面进行约束，因无法对其使用可视化方法来获得光滑且连续的网格。

由于 NeRF 惊艳的表现，越来越多基于 NeRF 的方法被提出，常远等人[15]和朱方[16]分别对 NeRF 的近期成果和发展方向进行了总结。一些研究者尝试将 NeRF 应用于表面重建任务。Niemeyer 等人[17]提出了一种在不依赖显式 3D 监督的情况下学习 3D 对象的表面表示的方法。该方法首先训练神经网络来预测一组 2D 图像的相机位姿。随后以其为输入，训练不同的神经网络来预测对象在 3D 空间中的占用情况。Yariv 等人[18]提出了一种端到端的神经网络架构，无需显式的表面提取即可使用隐式神经表示生成 3D 对象的表面。该方法使用 SDF 来描述空间点到 3D 对象表面的 SDF 值的映射关系，通过执行体积渲染从带掩码的图像中学习 3D 对象的几何形状、表面纹理和相机曝光参数。Jiang Yue 等人[19]提出了一种全新的基于 SDF 的可微渲染器，它能够同时计算形状参数在不同分辨率下的梯度，这提高了反向传播的效率和稳定性。Oechsle 等人[20]首先总结了从 2D 图像中恢复 3D 对象的两种不同的任务：表面重建和新视角合成。随后，提出了一个能够同时执行上述两种任务的统一框架，它融合了表面表示和体积表示各自的技术优势。Wang Peng 等人[21]提出了一个能够对 SDF 进行无偏估计的体积渲染公式。该方法不依赖表面上的纹理特征，能够在无掩码的情况下对具有部分遮挡或复杂结构的 3D 对象进行高质量的表面重建。

1.3 研究内容与方法

针对当前主流的隐式神经表面重建方法仍然存在的问题与挑战，本文以提升现有方法的重建精度为目标，分别从基于点云的表示和基于图像这两个方面展开了深入的研究，提出了相应的几点改进，达到了预期的研究目标。

对于基于点云的隐式神经表面重建方法难以精确的恢复 3D 对象复杂的表面纹理和拓扑结构的问题，以提高输入数据的信息量为切入点，研究了如何在隐式神经表示中有效的利用坐标邻域内的面特征。在不增加采样规模的前提下，以降低模型对复杂的表面细节的拟合难度并提高了模型对输入数据的利用率为目的，提出了一种全新的改进模型来改善表面重建的质量。此部分的主要研究内容如下：

（1）对于现有的基于点云的隐式神经表面重建方法进行了总结，指出现有方法目前仍然存在的不足及其产生的原因：仅使用了坐标信息，未考虑到其邻域表面上的特征对局部形状的贡献。

（2）设计一种使用编码-解码策略的隐式神经表示模型。编码器用于获取坐标在邻

域表面上的特征编码和其在高维空间下的映射编码。解码器用于学习 3D 对象的几何表面。

（3）设计一种自适应的损失调节策略，通过动态调节优化的损失，从而提高模型对输入的编码信息的利用率。

针对基于图像的隐式神经表面重建方法难以对高精度的重建复杂的 3D 对象，并且其表面不够光滑等一系列问题，以提高模型对噪声数据的抗干扰能力为目的，设计了一种全新的网络架构，通过引入新的归纳偏执来提高了表面重建的精度。该部分的主要研究内容如下：

（1）对于现有的基于图像的隐式神经表面重建方法进行总结，指出现有方法目前仍然存在的不足及其产生的原因：使用位置编码[14]对坐标进行高频映射时产生了大量高频噪声，降低了模型的鲁棒性。

（2）设计一种带静态注意力机制的网络架构，通过使用门控单元和线性映射操作来引入新的归纳偏执，从而提高表面重建结果的质量和光滑度。

（3）设计一种体素结构来粗略的划分空间区域，通过在训练过程中动态更新每个子体素内的 SDF 值，来规避对高 SDF 值的空间区域的采样点进行训练，从而提高训练效率。

（4）将球谐函数（spherical harmonic function）引入模型，通过显式的计算非朗伯体在不同视角下的光线反射情况，来降低模型预测的难度，并提高表面重建的稳定性。

1.4 章节安排

本文总共分为五个章节，每个章节的内容如下：

第一章为绪论。首先，阐述了使用隐式神经表示进行表面重建的研究背景与意义。其次，按照以点云为输入和以图像为输入的划分方式，分别介绍了国内外隐式神经表面重建的研究现状。随后，分别分析了上述两种研究方向目前所面临的挑战与困难。最后，总结了本文的研究内容、研究方法和主要贡献。

第二章是对相关技术与背景知识的介绍。首先，介绍了一些传统的表面重建方法，并分析了它们的优缺点。其次，详细解释了隐式神经表面重建中的一些专业术语和理论知识，为后续的研究内容提供理论支持和参考。最后，提供了在定量实验中使用的评价指标及计算公式，以便读者更好地进行比较。

第三章提出了一种基于点云的隐式神经表面重建的改进方法。首先，分析了目前现有方法 DeepSDF 存在的不足和改进研究的方向。接着，提出了一种编码—解码的改进模型来解决现有方法存在的问题，并对其各模块的作用进行了详细阐述。随后，在形状重建实验中与现有方法进行了定性和定量对比，并通过消融研究来验证每个改进模块的有效性。最后，测试了改进模型在单视角输入重建应用和形状编辑应用的表现，进一步

验证了其强大的性能表现。

第四章提出了一种基于图像的隐式神经表面重建的改进方法。首先，简要介绍了注意力机制的原理和发展，引出了使用注意力机制来改进现有方法 Neus 的技术路线。其次，提出了一种使用注意力模块来代替传统全连接神经网络的新方法，并详细阐述了该注意力模块的理论依据和实现细节。接着，为了进一步优化模型性能，还应用了球谐函数和采样点剪枝策略。随后，在形状重建实验中与现有方法进行了定性和定量对比，并通过消融研究来验证每个改进模块的有效性。最后，通过设计实验来验证注意力模块在其他框架中的泛化能力。

第五章为总结与展望。首先，全面总结了研究工作，包括方法的设计、实验的过程和结果的分析，同时详细阐述了这些工作的优点和不足。其次，通过对现有研究的局限性和问题的深入分析，本文探讨了未来可能的研究方向，并提出了一些创新性的想法和建议，为相关领域的后续研究提供了有价值的参考。

2 背景知识和相关技术介绍

2.1 传统的表面重建方法综述

2.1.1 显式方法

德劳内三角剖分（Delaunay triangulation）[22]是一种几何概念，用于对平面或高维空间中一组点进行三角化。德劳内三角剖分满足唯一性和最优性准则，使其被广泛应用于计算机图形学、计算几何和有限元分析等各个领域[23-25]。德劳内三角剖分具有空圆性质、最大化最小角性质和最小化外接圆等优良性质，即三角剖分的任何三角形的外接圆内不包含点集中的任何顶点；三角剖分最大化了每个三角形中的最小角，避免了长而细的三角形；三角剖分最小化了每个三角形的最大外接圆半径，提供了良好的三角形分布。因此其三角剖分的数值精度和网格质量比其他同类型的算法更优秀。德劳内三角剖分有多种计算方法，包括增量算法、分治算法和扫描线算法。这些算法通过递增地、递归地或以扫描线方式添加点来构建三角剖分。

沃罗诺伊图（Voronoi diagram）[26]是一种图结构，用于根据到一组种子点的距离将平面或高维空间划分为多个区域，所构成的图又称泰森多边形。对于每个种子点，都有一个相应的区域，称为沃罗诺伊单元，由空间中所有比到任意其他种子点距离更近的点组成。沃罗诺伊图的一个重要属性是它与德劳内三角剖分的对偶关系。具体来说，沃罗诺伊图的顶点是德劳内三角形的外心，沃罗诺伊图的边是德劳内三角形一边的垂直平分线。这种对偶关系在许多应用中都很有用，例如计算流体动力学和网格生成[27, 28]。构造沃罗诺伊图的算法有几种，包括蛮力算法、增量算法、分治算法和 Fortune 算法。每种算法都有其优点和缺点，具体取决于种子点的数量和应用场景。

2.1.2 隐式方法

Hoppe 等人[29]提出了一种隐式的表面重建算法，能够从无序的点云中建立一个隐式函数来获取任意坐标点到 3D 对象表面的 SDF 值，最后通过提取该函数的零等值面得到 3D 对象的近似表面。为了解决传统的表面重建方法需要根据具体情况来进行设计的局限，该方法将所有特定应用场景下的表面重建抽象为一个一般问题，不对数据的结构做任何假设。该算法建立了隐式表面重建的基本框架，具有极高的理论和实践价值。

Curless 等人[30]提出了 VRIP 算法（volumetric range image processing），它认为测距图的测量误差为沿着投影方向服从高斯分布，并将其成了一个最大似然估计问题。VRIP 算法一次只处理一张测距图像，并通过累积加权有 SDF 来更新 3D 对象的体积表示。首先，将其深度值转换为距离函数。随后，使用简单的加法方案将其与已经获得的数据相结合。为了实现空间效率，采用体积的游程编码。为了实现时间效率，对测距图像进行

重新采样，以便与体素网格对齐，并同时遍历测距线和体素扫描线。最后，通过从体积网格中提取等值面来生成最终的流形网格。该算法具有增量式更新、在方向不确定时运行、填补表面的空隙和对异常值的鲁棒性等优点。

Carr 等人[31]使用多项式径向基函数（radial basis function，RBF）的加权组合来代替拟合切平面计算 SDF，能够从点云中重建平滑、连续的流形表面，并修复不完整的网格。3D 对象的表面由 RBF 的零点集来隐式地定义。该方法在拟合过程中使用贪心算法，减少了表示表面所需的 RBF 中心数量，从而显著压缩了计算量。通过最小化多项式径向基样条的能量可以获得平滑的表面插值结果。这种与尺度无关的特性非常适合从非均匀采样的数据中重建表面，并且能够平滑地填补空洞。

Kolluri 等人[32]提出了隐式移动最小二乘算法（Implicitly Moving Least Squares，IMLS），从理论上验证了其结果是原始表面的 SDF 的良好近似，并且在几何和拓扑上是正确的。首先，为每个空间点定义一个全局平滑点函数，来近似其局部邻域中的 SDF 值。随后，使用高斯权重函数混合所有平滑点函数，获得一个平滑的隐函数，其零水平集为重建的流形表面。

Kazhdan 等人[33]提出了泊松表面重建算法（Poisson Surface Reconstruction，PSR），它能够从无组织的点云中重建出光滑且水密表面。该算法将表面的梯度视为矢量场来求解泊松方程，并使用梯度场的无发散特性来增强表面的平滑度。首先，计算点云的 SDF 并用它来估计表面的梯度。然后，建立泊松方程来求解梯度场，得到体积标量场。最后，通过提取标量场的零等值面，得到平滑的流形表面。泊松曲面重建算法可以更好地保留细节信息，能够在处理噪声数据和稀疏的点云时产生更加精确的结果。但是，其需要计算大量的梯度和拉普拉斯算子，因此对计算和内存资源的要求较高。近些年，许多研究者一直致力于对其进行轻量化的改进[34,35]。

2.2 隐式神经表面重建的相关技术

（1）SDF

SDF 是一种数学表示法，它测量从空间中的点到物体表面的距离，同时考虑表面法线的方向。SDF 描述了一个标量场，其中空间中的每个点都被分配了一个带符号的距离值，表明它与物体表面的接近程度。距离的符号表示该点是在对象内部还是外部，而距离的大小表示到表面的最短距离。

给定三维空间中一个空间区域Ω，任意坐标点$x \in \mathbb{R}^3$对应的 SDF 值可以表述为：

$$d(x)=\begin{cases}\min\limits_{y\in\partial\Omega}\|x-y\|_2, & x\in\Omega\\ -\min\limits_{y\in\partial\Omega}\|x-y\|_2, & x\notin\Omega\end{cases} \tag{2.1}$$

式中：$\partial\Omega$ 表示空间区域 Ω 的边界处；$y \in \mathbb{R}^3$ 表示边界 $\partial\Omega$ 上的一点。

如果边界 $\partial\Omega$ 是平滑的，则其 SDF 在任何地方都是可微的，满足梯度方程：

$$|\nabla d| = 1 \tag{2.2}$$

因此保证了对其进行离散化后的表面为流形。总而言之，SDF 以一种简单而有效的方式表示复杂对象，能够轻松地判断点与表面之间的关系，是隐式表示中最为主流的表示方法之一。

（2）步进立方体算法

步进立方体（Marching Cubes，MC）[36]是一种从空间标量场中提取 3D 对象表面网格的算法。该算法通常以一组 3D 点作为输入，每个点都在标量场中有一个与之关联的量化值。标量场可以表示在 3D 空间中不同位置的各种物理量，例如：密度、颜色或距离。该算法通过划分空间来创建一组用来近似标量场表面的多边形网格，常被用于 3D 可视化。

MC 算法将 3D 空间划分为具有固定数量的立方体网格，其中每个立方体包含八个顶点。首先，计算每个顶点在标量场中的量化值，然后通过将其与一个规定阈值进行比较，确定标量场表面是否与该立方体相交。随后，如果标量场表面与某个立方体相交，则将该立方体细分成 8 个更小的子立方体，并通过从预先制定的配置查找表中选择一个最接近的多边形来近似标量场的局部表面。每个立方体网格共有 256 种可能的多边形配置，每个配置由子立方体顶点的唯一组合表示。

表面网格的质量取决于网格的分辨率和用于近似每个立方体内表面的子立方体的大小。MC 算法非常消耗计算资源，需要借助分层结构和强大的并行计算设备来应对大规模数据的处理。由于立方体数量按指数级的增长，该算法在处理复杂标量场时可能会遇到内存限制和计算速度缓慢的问题。

（3）Alpha 颜色混合

Alpha 颜色混合[37]是一种常用于计算机图形学和计算机游戏开发中的图像渲染技术，用于将一个图像或物体叠加在另一个图像或物体上，以实现透明度和半透明效果。在 Alpha 颜色混合中，每个像素包含一个额外的 Alpha 通道，该通道表示该像素的透明度，其计算公式可以表示为：

$$c = c_f \cdot \alpha + c_b \cdot (1 - \alpha) \tag{2.3}$$

式中：c_f 和 c_b 分别表示前景和背景像素值；α 表示混合比例；c 表示混合后的像素值。

在混合过程中，每个像素的颜色值和 Alpha 值都会参与计算，通过调整 Alpha 值可以改变图像的混合比例，从而改变混合后的效果。如果两个像素的 Alpha 值相加小于 1，

那么它们的颜色值将按照它们的 Alpha 值加权混合，得到一个介于它们两者之间的新颜色值。如果两个像素的 Alpha 值相加大于等于 1，那么它们的颜色值将以一定的方式进行合并，以确保颜色的总体亮度和饱和度得到正确的处理。

（4）光线投影原理

NeRF 和相关方法使用逆向渲染技术从一组 2D 图像中估计出一个 3D 物体或场景的表示。其中，一个重要的前置步骤是获取图像的相机位姿。相机通过采集特定视角下 3D 对象的投影光线来生成 2D 图像，这种投影关系是一种单射函数。由于深度信息的丢失，从 2D 图像中逆向恢复 3D 信息非常的困难。

相机的投影模型有很多种，其中最简单和最常用的是一种线性近似模型—针孔模型，它通过基本的截距定理来定义任意 3D 点投影到图像平面上正确的位置，可以用一个线性变换来描述：

$$K=\begin{bmatrix}\alpha_x & \gamma & c_x & 0\\ 0 & \alpha_y & c_y & 0\\ 0 & 0 & 1 & 0\end{bmatrix} \tag{2.4}$$

式中：K 表示相机的内参矩阵；$f=\left[\alpha_x,\alpha_y\right]$表示像素大小归一化的焦距；γ 表示相机的同轴度；$c=\left[c_x,c_y\right]$表示相机平面的中心位置。

假设某个 3D 点的归一化坐标为 $p'=\left[x,y,z,1\right]^{\mathrm{T}}$，则其在图像上的投影可以表示为：

$$q'=K\cdot p' \tag{2.5}$$

式中：$q'=\left[u,v,1\right]^{\mathrm{T}}$ 为归一化的像素坐标。

为了在多个不同的相机之间进行坐标统一，需要将它们的相机坐标系转换为同一个参考系，通常选择其中一个相机的位置作为世界坐标系原点，并通过应用一个刚体变换来对齐不同相机坐标系之间的坐标系，这个刚体变换就被称为相机的外参矩阵 T，可以表示为：

$$T=\begin{bmatrix}R_{3\times3} & t_{3\times1}\\ 0_{1\times3} & 1\end{bmatrix} \tag{2.6}$$

式中：R 表示一个旋转矩阵；t 表示一个平移矢量。

坐标系的转换过程可以描述为：

$$p_c=R\cdot p_w+t \tag{2.7}$$

式中：p_w 表示世界坐标下的一个 3D 点，p_c 表示相机坐标下的一个 3D 点。

为了准确建模相机，必须考虑相机畸变的影响。一般情况下，相机畸变模型包含多

达 12 个畸变参数。在针孔模型中，通常只使用 3 个径向畸变参数和 2 个切向畸变参数来进行简化建模，因此也导致了整个相机投影过程的不可逆。

（5）体积渲染

体积渲染以光线投影为基础，并已被证明在神经渲染中是有效的，特别是在从多视图输入数据中学习场景表示方面。具体来说，它使用连续的体积密度或占用率来表示场景，而不是硬表面的集合。这意味着射线在空间的每一点上都有与场景内容相互作用的一些概率，即光线可能被其路径上的每一个体积粒子吸收或者反射。这种连续的表示能够很好地利用基于深度学习的可微分渲染框架，在很大程度上利用良好的梯度来进行优化。

给定一个任意的像素点，可以通过针孔相机模型将其逆向转换为一条从相机光心发射，并且穿过整个目标空间区域的光线，其在世界系下的原点和方向分别为 p_o 和 d_o。这条入射光线在路径中被体积介质吸收或反射的情况可以被简单定义为：

$$L\left(p_o,d_o\right)=\int_{t_0}^{t_1}T\left(p_o,d_o,t_0,t\right)\sigma\left(p_o+td_o\right)L_e\left(p_o+td_o,-d_o\right)dt \tag{2.8}$$

式中：t_0 和 t_1 分别表示光线传播的起点和终点；t 表示光线当前传播的距离；$\sigma(\cdot)$ 表示光路经过的某个位置的体积密度，即与体积介质交互作用的概率；$L_e(\cdot)$ 表示光路经过的某个位置在其传播方向下与体积介质交互作用的结果；$T(\cdot)$ 表示透射率。

透射率的积分表达式为：

$$T\left(p_o,w_o,t_0,t\right)=\exp\left(-\int_{t_0}^{t}\sigma\left(p_o+sw_o\right)ds\right) \tag{2.9}$$

式中：$s\in[t_0,t]$ 表示当前已经过的路径，它描述了光线从相机出发，到达某一点 p_o+td_o 的过程中与目标空间区域内的体积介质发生交互作用后的衰减程度。

在实践中，通过使用正交法对上述积分进行离散化，其中 σ 和 L_e 均被假设为在局部区域内是恒定的，并通过对光路进行间隔采样来近似估计[38]。假定使用一组等距的区间 $\left[t_{i-1},t_i\right)_{i=1}^{N}$ 来分割光线，则其离散化的积分过程可以表示为：

$$\begin{aligned}L\left(p_o,\omega_o\right)&\approx\sum_{i=1}^{N}T_i\alpha_iL_e^{(i)}\\T_i&=\exp\left(-\sum_{j=1}^{i-1}\Delta_j\sigma_i\right)\\\alpha_i&=1-\exp\left(-\Delta_i\sigma_i\right)\end{aligned} \tag{2.10}$$

式中：$\Delta_i=t_i-t_{i-1}$ 表示采样间隔；α_i 则可以与 Alpha 颜色混合中的混合比例相对应。

（6）基于坐标的多层感知机

神经网络是当前机器学习领域广泛应用的技术之一，可以被用于图像识别、语音识别和多模态等多种领域。神经网络是一种高度并行的信息处理系统，具有极强的自适应学习能力，不需要研究对象的数学模型，能够处理复杂的非线性系统，并且具有很好的鲁棒性，能够应对系统参数变化和外部干扰的影响。

多层感知机（multi-layer perceptron，MLP）是一种常用的神经网络结构，包括输入层、隐藏层和输出层，其中相邻层之间的神经元是全连接的。MLP 主要包括三个基本要素：权重、偏置和激活函数。权重表示神经元之间的连接强度，它们的大小决定了输入信号在网络中传递的可能性大小。偏置用于保证通过输入计算出的输出值不能被轻易地激活。激活函数起到非线性映射的作用，将神经元的输出幅度限制在一定范围内，以便更好地处理输入信号。

众所周知，MLP 可以作为通用的函数逼近器[39]。近年来，越来越多的研究者尝试使用 MLP 拟合一个连续的函数来替代传统的离散表示，用于 3D 场景表示。MLP 以几何空间中的一组坐标作为输入，并输出与坐标相对应的一些量化值，这种网络也被称为基于坐标的神经网络，其所生成的场景表示称为基于坐标的场景表示。需要注意的是，输入的坐标空间可以与欧几里得空间对齐，也可以是低维空间的嵌入，例如网格的 UV 空间。

（7）注意力机制

注意力模型（Attention Model）是一种广泛应用于深度学习任务中的核心技术，包括自然语言处理、图像识别和语音识别等。其基本思想是通过计算模型对输入信息的重要性，将有限的计算资源集中用于更重要的任务，从而提高模型的效率和准确性。

注意力机制（Attention Mechanism）是实现注意力模型的关键组成部分。它可以让模型自己学习如何分配注意力资源，将有限的注意力资源集中在对当前任务更为关键的信息上，同时忽略无关信息。这种机制与人类的视觉注意力机制类似，通过选择关键信息提高处理效率和准确性。

注意力机制分为软注意力、强注意力和自注意力机制三类。软注意力通过给输入信息加权来突出重要的特征。强注意力则是在输入信息中选择最重要的部分。自注意力机制则是在序列数据中学习每个位置的重要性，并为每个位置分配不同的权重。

总之，注意力模型和注意力机制是深度学习技术中非常重要的组成部分，可以有效解决信息过载和提高任务处理效率和准确性。

2.3 评价指标

2.3.1 网格生成任务的评价指标

本文采用 Tretschk 等人[8]的误差指标计算方法，通过重叠度（Intersection over Union，IoU）、倒角距离（Chamfer Distance，CD）和 F-Score 这些指标来评价网格模型的重建质量。

IoU 是一个用于评估两个集合重叠程度的指标，可以用于简单且直观的评估重建模型与真实模型之间的相似度，该指标越高越好。具体而言，它通过计算两个点云之间相交的体积与它们并集的体积之比来衡量它们的重叠度。其计算公式为：

$$d_{\mathrm{IoU}}=\frac{\Omega_1 \bigcap \Omega_2}{\Omega_1 \bigcup \Omega_2}\cdot 100\% \tag{2.11}$$

式中：Ω_1 和 Ω_2 分别表示在重建模型与真实模型的内部空间区域。

CD 是一个用于评估两个点云分布接近程度的指标，可以用于评估重建模型与真实模型局部表面的平均误差距离。该指标越低越好。具体来说，以相同的策略在重建模型和真实模型的表面上进行采样。对于重建模型的表面点云中的每个点，在真实模型的表面点云中找到其最临近点，得到重建模型到真实模型的平均距离；同样的，反向计算真实模型到重建模型的距离。最后，取二者距离之和的平均值得到 CD。其计算公式为：

$$d_{\mathrm{CD}}=\frac{1}{S_1}\sum_{p\in S_1}\min_{q\in S_2}\|p-q\|_2+\frac{1}{S_2}\sum_{p\in S_2}\min_{q\in S_1}\|p-q\|_2 \tag{2.12}$$

式中：S_1 和 S_2 分别表示重建模型与真实模型表面上的采样点云；p 和 q 分别表示来自两个不同点云的采样点。

F-score 是一种用于评估二分类问题的指标，它是精确度（precision）和召回率（recall）的调和平均数，可以用于评估重建模型与真实模型之间的重叠程度，该指标越高越好。具体来说，真实 3D 模型作为正例（positive），将重建的 3D 模型作为负例（negative）。那么，对于一个重建模型，可以计算出它与真实模型的重叠部分的体积（true positive，TP）、它独有的体积（false positive，FP）和真实模型独有的体积（false negative，FN）。其计算公式为：

$$\begin{aligned} d_{\mathrm{F-Score}}&=\frac{2\cdot \mathrm{precision}\cdot \mathrm{recall}}{\mathrm{precision}+\mathrm{recall}}\cdot 100\% \\ \mathrm{precision}&=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \\ \mathrm{recall}&=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \end{aligned} \tag{2.13}$$

2.3.2 图像生成任务的评价指标

本文采用 Mildenhall 等人[14]的误差指标计算方法，通过峰值信噪比（Peak Signal to Noise Ratio，PSNR）、结构相似性指数（Tructural Similarity Index，SSIM）和学习感知图像块相似度（Learned Perceptual Image Patch Similarity，LPIPS）这些指标来评价图像的重建质量。

PSNR 是一种常用的评价图像质量的度量标准，通过计算感兴趣信号与背景噪声的强度比，来衡量图像的失真情况，该指标越高越好。PSNR 将两张像素大小为 $m\times n$ 的图像的均方误差（Mean Square Error，MSE）转换成了某种对数表示的形式，单位为分贝（dB）。其计算公式为：

$$\begin{aligned} d_{\mathrm{PSNR}} &= 10\cdot\log_{10}\left(\frac{2^n-1}{\mathrm{MSE}}\right) \\ \mathrm{MSE} &= \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left(I_1(i,j)-I_2(i,j)\right)^2 \end{aligned} \tag{2.14}$$

式中：I_1 和 I_2 分别表示输入图像与参考图像；n 表示图像的比特数（bit），通常取值为 8。

通常来说，当 PSNR 高于 40dB 时，图像质量几乎与原始图像一样好；当 PSNR 在 30-40dB 之间时，通常表示图像质量的失真损失在可接受范围内；当 PSNR 在 20-30dB 之间时，说明图像质量相对较差；而当 PSNR 低于 20dB 时，则说明图像失真程度相当严重。需要注意的是，PSNR 作为一种图像质量评价指标，虽然可以简单地量化图像质量，但并不一定能完全反映人类视觉对图像质量的感知，因为它忽略了人眼感知的差异性。

SSIM 是一种用于测量两张图像相似性的指标，它考虑了亮度、对比度和结构三个方面的信息，可以更好地反映人眼对图像质量的主观感受，该指标越高越好。具体来说，首先对比输入图像和参考图像的亮度值，用高斯滤波器平滑后，计算均值和标准差，再计算亮度相似性 l；其次对比输入图像和参考图像的对比度值，用高斯滤波器平滑后，计算标准差，再计算对比度相似性 c；最后对比输入图像和参考图像的结构信息，通过对输入图像和参考图像进行亮度和对比度的归一化，计算它们之间的协方差，最终得到结构相似性 s。其计算公式为：

$$
\begin{aligned}
l\left(I_1, I_2\right) &= \frac{2\mu_1\mu_2 + c_1}{\mu_1^2 + \mu_2^2 + c_1} \\
c\left(I_1, I_2\right) &= \frac{2\sigma_1\sigma_2 + c_2}{\sigma_1^2 + \sigma_2^2 + c_2} \\
s\left(I_1, I_2\right) &= \frac{2\sigma_{12} + c_3}{\sigma_1 + \sigma_2 + c_3}
\end{aligned} \tag{2.15}
$$

式中：μ_1 和 μ_2 分别为 I_1 和 I_2 的平均值；σ_1 和 σ_2 分别为 I_1 和 I_2 的标准差；σ_{12} 为 I_1 与 I_2 的协方差；c_1、c_2 和 c_3 为三个用于维持计算稳定的微小的常数，且 $c_2 = 2c_3$。

SSIM 将亮度相似性、对比度相似性和结构相似性三个方面的相似性综合起来，得到一个 0 到 1 之间的评分。其计算公式为：

$$
d_{\mathrm{SSIM}} = l^{\alpha} \cdot c^{\beta} \cdot s^{\gamma} \tag{2.16}
$$

式中：α、β 和 γ 分别为比重控制参数，一般均被设置为 1。

LPIPS 是一种用于衡量图像质量的指标，通过计算两个图像在感知空间中的距离来评估它们之间的相似性，该指标越低越好。与其他传统的图像质量评估指标（如 PSNR 和 SSIM）不同，LPIPS 使用了基于深度学习的方法来模拟人类视觉系统的感知过程。具体来说，LPIPS 首先通过一个经过预训练的卷积神经网络（通常是 VGG 网络）对输入的图像进行特征提取，然后计算两个图像在特征空间中的欧氏距离。为了提高指标的鲁棒性，LPIPS 还对特征进行了标准化和对齐等处理。LPIPS 已经在多项图像质量评估任务中证明了其优越性，包括超分辨率、去模糊、风格转移等。同时，LPIPS 还具有一定的通用性，可以适用于不同分辨率、不同颜色空间的图像，以及不同类型的失真（如压缩、噪声等）。

2.4 本章小结

本章节深入介绍了一些相关技术和背景知识，分为三个部分。第一部分，介绍了几种传统的显式和隐式表面重建方法，并对它们的适用范围、精度和计算效率等方面进行了简要分析。这些方法包括点云重建、网格生成和基于隐式函数的表面重建。第二部分，详细介绍了隐式神经表面重建中各个环节所涉及到的技术细节，并解释说明了其原理和公式。这些技术包括符号距离函数、步进立方体算法、Alpha 颜色混合、光线投影原理、体积渲染技术、基于坐标的多层感知机和注意力机制等。第三部分，介绍了网格生成任务和图像生成任务中常用的评价指标，包括 IoU、CD、F-Score、PSNR、SSIM 和 LPIPS，并给出了相应的计算过程。

3 基于点云的隐式神经表面重建的改进

针对现有基于点云的隐式神经表面重建方法仍然难以实现对复杂的 3D 对象的高精度重建的问题，以提高输入数据的信息量为切入点，研究了如何在隐式神经表示中有效的利用坐标邻域内的面特征。不增加采样规模的前提下，以降低模型对复杂的表面细节的拟合难度并提高了模型对输入数据的利用率为目的，设计了一种采用编码－解码策略的改进模型。编码器用于获取坐标在邻域表面上的特征编码和其在高维空间下的映射编码。解码器用于学习 3D 对象的几何表面，同时对其优化器的损失进行了动态加权。

3.1 改进模型

改进模型基于 DeepSDF，它将三维形状表示为由神经网络参数化的隐式函数，该函数可以将空间中的任意点映射到距离最近的表面上的 SDF 值。DeepSDF 使用了一种名为 Auto-Decoder 的网络架构，该架构仅由多个全连接层和 ReLU 激活层组成，如图 3.1 所示。这种设计简单地架构易于与其他深度学习模块进行集成，使得改进的实施和有效性验证更加容易。改进后的模型输入和输出协议与 DeepSDF 保持一致，以确保兼容性和可靠性。

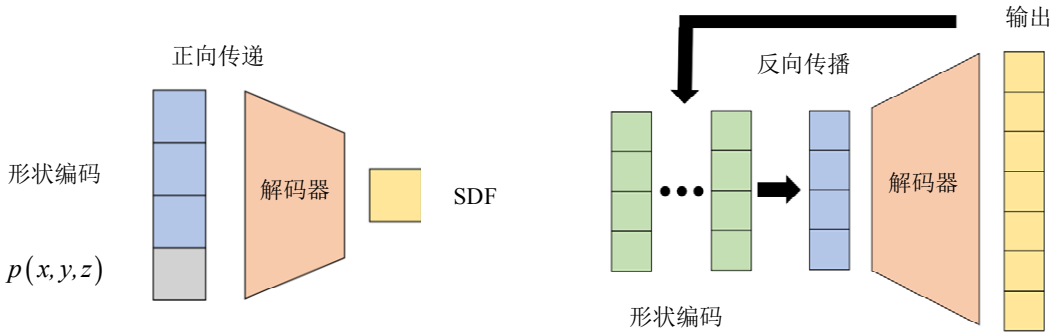


图 3.1 DeepSDF 的原始架构

Figure 3.1 The original architecture of DeepSDF

改进模型将 3D 对象表面表示成一个连续 SDF。该函数描述了任意空间坐标 $p \in \mathbb{R}^3$ 与该位置到 3D 对象表面最近处的有 SDF 值 $s \in \mathbb{R}$ 之间的映射关系。改进模型由编码器和解码器两个部分组成，可以表示为：

$$\begin{aligned} \left(\gamma(p), f\right) &= F_{\mathrm{EN}}(p) \\ s &= F_{\mathrm{DE}}\left(\gamma(p), f\right) \end{aligned} \tag{3.1}$$

式中：$F_{\mathrm{EN}}(\cdot)$ 表示编码器；$F_{\mathrm{DE}}(\cdot)$ 表示解码器；f 表示邻域表面特征编码；$\gamma(\cdot)$ 表示高频编码。

改进模型的整体架构如图 3.2 所示，编码器分别通过提取坐标在邻域表面内先验信息和对坐标值进行傅里叶变换来进行编码。解码器从编码信息中学习表面的几何特征，并对其 SDF 进行拟合。此外，解码器为每个 3D 对象训练一个独立的形状编码，它需要与前馈网络共同配合使用。训练好的模型可以被认为是一个以 3D 对象表面的零等值面为决策边界的二元分类器，它能够预测任意查询点所对应的 SDF 值。通过使用 MC 算法能够从其中显式提取出完整的网格模型。

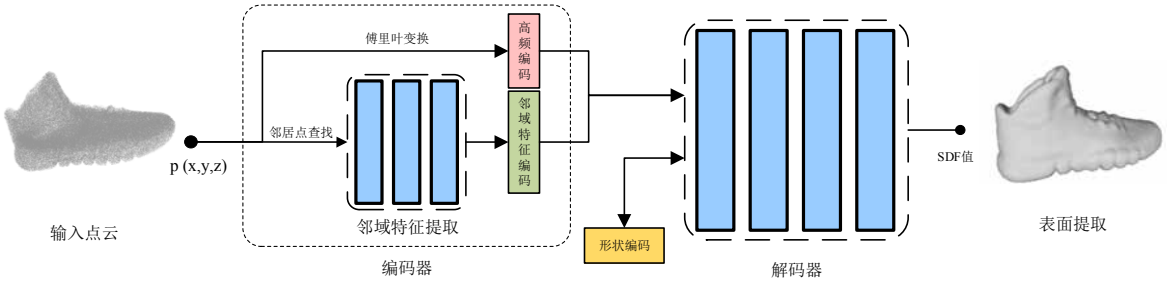


图 3.2 改进模型的整体架构

Figure 3.2 The overall architecture of the improved model

3.2 编码器

基于 SDF 的隐式神经表示方法针对空间坐标与 SDF 值之间的单射关系进行了建模，每个采样点的预测 SDF 值在 3D 对象表面上对应一个唯一的位置。但是，每个表面位置可能由多个采样点的预测 SDF 值的共同作用下决定，这种多对一的情况在 3D 对象非平滑的局部区域中尤为突出。因此，现有方法在复杂的局部表面上非常容易受到少量噪声的干扰，难以逼真的恢复精细的表面纹理和复杂的拓扑结构。

为了解决此问题，在表面拟合之前，对输入的坐标进行编码，以减轻下游的解码器的负担。首先，在每个采样点的邻域表面上进行邻居点查找。其次，使用一个神经网络从邻居点的坐标和法向量中提取该局部区域的几何先验。最后，使用傅里叶变换将采样点的坐标分解成多个频率基，并随着训练的进行不断增加其数量。

3.2.1 邻居点查找

改进模型应用了 DeepSDF[2]的数据预处理方法，在开源数据集中的网格上采集训练模型所需的数据。该方法首先在 3D 对象的表面上进行随机的坐标采样，随后对其进行两次随机扰动产生两个新采样点。按所在空间区域的不同，采样点可以被划分为：内部点（SDF 值为负）、外部点（SDF 值为正）和表面点（SDF 值为零）。编码器使用表面点作为描述邻域表面信息的邻居点。为增强邻居点携带的邻域信息，在采样过程中为其额外的估计了法向量。使用 K-邻近算法（KNN）为每个采样点查找 K 个最邻近的表面点 $x_1 \sim x_K$，并将其作为该采样点的邻居点。如图 3.3 所示，曲线表示 3D 对象在某局部位

置的表面。为了表示的简洁，仅画出了任意采样点 p 的 3 个邻居点 x_1、x_2 和 x_3。

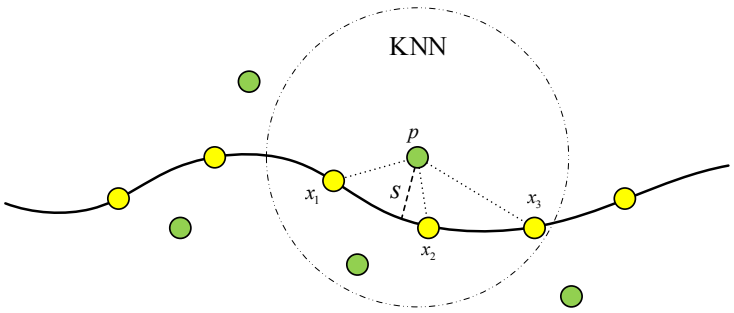


图 3.3 在局部表面上查找邻居点

Figure 3.3 Finding neighbors on a local surface

3.2.2 邻域表面特征编码

为了保证邻域表面特征编码的平移不变性，提高编码器的泛化能力，邻居点的坐标均采用以其采样点为坐标原点的相对值。邻域表面特征提取网络同时输入 K 个邻居点的相对坐标 $p-x_i$ 和法向量 n_i，输出一个邻域表面特征编码 f，该查找过程可以表示为：

$$\begin{aligned} f &= \sum_i \frac{1}{\left\| p-x_i \right\|_2^2} f_i \\ f_i &= G\left(p-x_i, n_i \right) \end{aligned} \tag{3.2}$$

式中：$i \in [1, \mathrm{K}]$；f_i 表示邻居点 x_i 的邻域表面特征分量；G 表示神经网络。

与 PointNet[40]的架构类似，邻域表面特征提取网络由一组串联的共享的 MLP 组成，如图 3.4 所示。PointNet 是一种处理点云的强大且有效的架构，它能够从不规则的数据结构中提取坐标点的特征信息，被广泛的应用在各种基于点云输入的应用中。与原始的 PointNet 不同，邻域表面特征提取网络使用了逆距离加权函数代替了最大池化操作，对网络的输出结果进行聚合。这保证了距离 p 越远的邻居点对 f 的贡献越小，能够在一定程度上避免在邻域范围边缘的噪声对邻域表面特征的干扰。

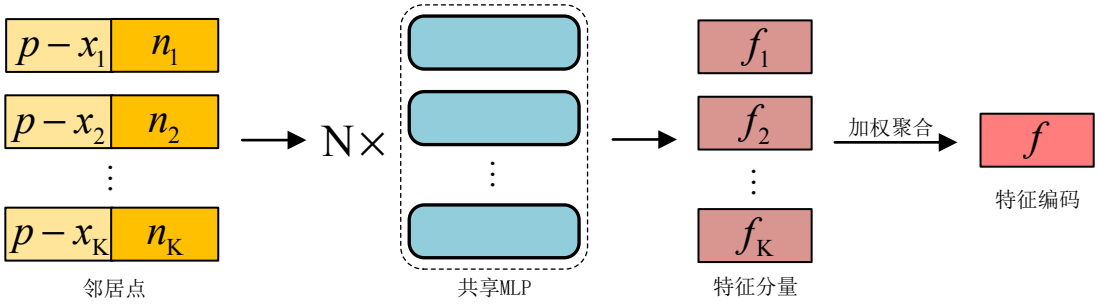


图 3.4 邻域特征提取网络架构

Figure 3.4 Neighborhood feature extraction network architecture

3.2.3 阶段性位置编码

Rahaman 等人[41]的研究表明，MLP 更倾向于先学习输入信号中的低频信息，随后逐渐尝试学习更高频率的特征。为了解决基于 MLP 的网络的局限性，Mildenhall 等人[42]提出了位置编码。为了更好地近似 3D 对象表面上的高频细节，位置编码使用傅里叶变换将输入的 3D 坐标分解成不同频率基上的分量集合 $\gamma(\cdot)$，其表达式为：

$$\gamma(p)=\left[p,\gamma_0(p),\cdots,\gamma_k(p),\cdots\gamma_{\mathrm{L}-1}(p)\right] \tag{3.3}$$

但是，这种高维映射同时也放大了输入数据中的噪声，需要各参数之间更为精确的配合。为此，使用了低通滤波器对位置编码输出的所有频率基上的分量进行平滑控制，其表达式为：

$$\gamma_k(p)=\omega_k\left[\cos\left(2^k\pi p\right),\sin\left(2^k\pi p\right)\right] \tag{3.4}$$

式中：$\gamma_k(\cdot)$为第 k 个频率基的编码分量；L 为频率基数；ω_k为第 K 个频率基的权重。

低通滤波器可以表示为：

$$\omega_k=\begin{cases}0, & \alpha<k\\ \dfrac{1-\cos\left((\alpha-k)\pi\right)}{2}, & 0\le\alpha-k\le1\\ 1, & \alpha-k>1\end{cases} \tag{3.5}$$

式中：$\alpha\in[0,\mathrm{L})$为低通滤波器的控制因子。

在训练初期，模型的预测偏差较大。低通滤波器会屏蔽所有频率基上的分量，即$\alpha=0$，来避免高频噪声的对模型稳定性产生的冲击。随着训练的进行，模型变得更有经验，低通滤波器将逐渐解锁更高频率基上的分量，直到所有的频率基均被激活，即$\alpha=\mathrm{L}$，来增强模型对高频细节的捕获能力。

3.3 解码器

解码器基于 Auto-Decoder[2]。解码器在一组由同类 3D 对象构成的数据集上进行训练，并且将表面的几何参数解耦成类别层面的形状先验和个体层面的形状编码。形状先验描述了所有同类 3D 对象共有的特征和元素，存储在解码器的网络参数中。形状编码描述了每个 3D 对象对共有特征进行重组的控制参数，存储在一个 256 维的潜向量中。

解码器单独计算了每个采样点的 SDF 预测值与监督值之间的误差损失，因此对采样点的规模和分布没有任何限制。在训练时，解码器通过同时优化所有的形状编码$\{z_i\}_{i=1}^{\mathrm{N}}$

和网络参数θ，来最大化所有同类 3D 对象表面的后验分布，其表达式为：

$$\arg\min_{\theta,z_i}\sum_{i=1}^{N}\sum_{j=1}^{M}L\left(D_\theta\left(z_i,\gamma\left(p_{ij}\right),f_{ij}\right),s_{ij}\right) \tag{3.6}$$

式中：$D_\theta(\cdot)$表示 Auto-Decoder；$L(\cdot)$表示损失函数。

在测试时，网络参数θ被固定，解码器通过快速的优化一个形状编码z来最大化某个 3D 对象表面的后验分布，其表达式为：

$$\arg\min_{z}\sum_{j=1}^{M}L\left(D_\theta\left(z,\gamma\left(p_j\right),f_j\right),s_j\right) \tag{3.7}$$

为了提高解码器对编码信息的利用率，在现有先进方法[6]的基础上进行了改进，提出了自适应损失加权策略。

3.3.1 自适应损失加权策略

为兼顾训练效率和模型性能，每个 3D 对象的采样点数通常被控制在 50 万个左右。这导致了 3D 对象在含有精细纹理和复杂结构的局部区域内的采样点非常少，难以充分覆盖上述区域的表面。由于缺乏先验信息，进行重复的补充采样也非常困难。在这种情况下，解码器即使对复杂的局部表面进行了错误的拟合，优化器也难以从微弱的平均损失中也找到正确的优化方向。为了充分利用在复杂的局部表面附近的采样点，在训练过程中根据预测偏差来判断该采样点的拟合难度，并对其损失进行自适应的加权。

SDF 值的符号有实际的物理含义，即两个 SDF 值符号不同的采样点分别位于 3D 对象水密表面的外部（正）和内部（负）。如果只使用 L1 范数来约束优化损失，即便在表面附近的某个采样点的预测偏差非常小，也有可能导致解码器对其 SDF 值的符号做出错误的判断，这将严重的影响到表面拟合的准确性。为此，按 SDF 值的预测偏差的不同，将采样点划分为高、中和低三种难度，如图 3.5 所示。SDF 值符号被错误预测的采样点被认为是高难度的；SDF 值的预测值偏小的采样点被认为是中难度的；SDF 值的预测值偏大的采样点被认为是低难度的。

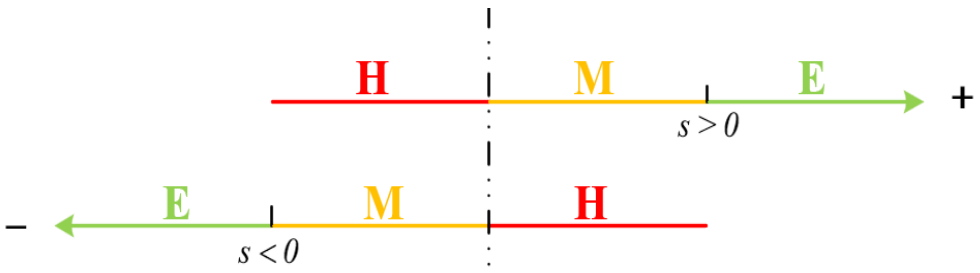


图 3.5 采样点的难度划分策略

Figure 3.5 Difficulty division strategy of sampling points

所有采样点的损失权重均被初始化为 1，并在前 500 个周期的训练中固定。这确保了在训练初期，每个采样点对优化方向产生相同的贡献。随着训练的进行，对预测偏差的惩罚力度将逐渐被注入到损失权重中。每经过 100 个周期的训练，每个采样点的难度被重新划分，并更新其损失权重。为了提高优化过程的稳定性，使用带惯性的权重更新方式能够使损失权重的变化过程更为平滑。其可以表示为：

$$\delta_{\text{new}} = f_{\text{tunc}}\left(f_{\text{sgn}}\left(s_{\text{p}}, s_{\text{t}}\right)\cdot\delta_{\text{old}}\right) \tag{3.8}$$

式中：s_{p} 和 s_{t} 分别为采样点的 SDF 的预测值和监督值；δ_{old} 和 δ_{new} 分别表示采样点在更新前后的损失权重；$f_{\text{tunc}}(\cdot)$ 为截断函数；$f_{\text{sgn}}(\cdot)$ 为控制函数。

为了使每个采样点都能对优化做出足够的贡献，使用截断函数对损失权重的区间进行限制，保证了个别极大的损失不会对优化方向起决定作用，其表达式为：

$$f_{\text{tunc}}(\tau) = \max\left(\tau_{\min}, \min\left(\tau_{\min}, \tau\right)\right) \tag{3.9}$$

式中：$\tau_{\max}$ 和 $\tau_{\min}$ 分别为损失权重的上下边界，分别取 1.6 和 0.6。

为了避免错误的预测 SDF 值的符号，高难度采样点的损失权重会持续增长，直到其预测偏差被纠正。为了更准确的从平均损失找到当前的优化方向，低难度样本的损失权重会持续下降。该控制函数的表达式为：

$$f_{\text{sgn}}\left(s_{\text{p}}, s_{\text{t}}\right) = \begin{cases} 1+\lambda, & s_{\text{p}}\cdot s_{\text{t}} < 0 \\ 1, & s_{\text{p}}\cdot s_{\text{t}} > 0 \cup s_{\text{p}}\left(s_{\text{t}} - s_{\text{p}}\right) \geq 0 \\ 1-\lambda, & s_{\text{p}}\cdot s_{\text{t}} > 0 \cup s_{\text{p}}\left(s_{\text{t}} - s_{\text{p}}\right) < 0 \end{cases} \tag{3.10}$$

式中：λ 为控制函数的参数，取值为 0.05。

3.4 损失函数

改进模型的整体优化损失函数可以表示为：

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{reg}} \tag{3.11}$$

式中：$\mathcal{L}_{\text{rec}}$ 和 $\mathcal{L}_{\text{reg}}$ 分别表示重建损失和正则化损失。

重建损失通过最小化 SDF 的预测值和监督值之间的误差来进行表面拟合，其表达式为：

$$\mathcal{L}_{\text{rec}} = \delta\left|f_{\text{tunc}}\left(s_{\text{p}}\right) - f_{\text{tunc}}\left(s_{\text{t}}\right)\right|, \tag{3.12}$$

式中：$f_{\text{tunc}}(\cdot)$ 表示截断函数与式 3.8 相同，其上下边界范围分别取值为 0.1 和-0.1。

正则化损失用于约束所有同类 3D 对象的形状编码在低维潜空间中服从高斯分布，增强了编码的泛化和插值能力，其表达式为：

$$\mathcal{L}_{\text{reg}} = \frac{1}{\sigma^2}\|z\|_2^2 \tag{3.13}$$

式中：σ 为正则化参数，取值为 0.01。

3.5 实验设置

3.5.1 数据集

ShapeNet[43]是一个 3D 合成模型数据集，它包含 16 个大类、共计 16881 个人工设计网格的大型 3D 形状合成数据集。从中选择了其中沙发、灯和飞机三个类别的网格作为实验数据。然而，该数据集中的网格存在大量未闭合的三角面，如果不进行预处理，将会导致大量异常 SDF 值的错误采样点。因此，采用了 Stutz 等人[44]的网格预处理方法，对实验数据中的所有网格进行了水密化处理，以在网格表面附近采集带有正确监督值的采样点。

Google Scan Objects（GSO）[45]是一个激光扫描的 3D 模型数据集，它由 1030 个从真实环境中采集的物品组成，包含 17 个类别。由于该数据集没有进行分类和网格预处理，因此首先对所有的网格进行了分类、筛选、姿态对齐和尺度归一化的处理，以满足数据集的尺度和旋转不变性。随后，从中选择了 253 个鞋类网格作为实验数据。

改进模型的一个输入样本由采样点、SDF 监督值和离该点距离最近的 K 个邻居点组成。首先，使用 OpenGL 在网格的单位球体表面上设置 100 个固定位置的虚拟相机来虚拟渲染每个相机视角下的深度图；通过相机参数反向投影所有的深度图中的像素点来采样 3D 对象表面，获得表面点云；其次，计算每个点所在三角面的法线方向并将其作为该点的法向量；然后，沿着 x、y、z 三个坐标轴方向，对每个采样点添加高斯噪声扰动来获得两个 SDF 符号相反的采样点；最后，使用 kD-tree 进行表面点的邻居点查找。

3.5.2 实施细节

为了保证比较的公平性，所有的实验遵循完全一致的训练与测试集划分和训练周期。使用 Adam 优化器[46]对数据集进行共计 2000 个周期的训练，批处理大小被设置为 32。邻域表面特征提取网络由三个串联的共享 MLP 组成，其通道数分别设置为 8、16 和 32。解码器网络由 8 个 256 个通道的全连接层组成，层间由 ReLU 进行激活并对权重归一化。在第四层中使用跳过连接操作来重新接入输入数据[2]。网络参数的学习率被初始化为 0.0005，并且每经过 500 个训练周期减半。形状编码的学习率被设置为 0.001。此外，所有的实验均是在一台配备 Nvidia RTX-3070 的服务器上个进行的。

3.6 分析与讨论

3.6.1 形状重建

为了验证改进的有效性，对 DeepSDF、Curriculum deepsdf（CurrSDF）与改进模型的完整版本及其几种消融版本在四个不同类别的数据集上进行了形状重建，定量结果如表3.1所示。表中最后一行为完整版本，第3-5行依次为只实施了邻域表面特征编码（FE）、阶段性位置编码（PE）和自适应损失加权策略（LW）的消融版本。

表3.1 形状重建的定量结果

Table 3.1 Quantitative results of shape reconstruction

IoU↑					
模型	Plane	Sofa	Lamp	Shoe	Mean
DeepSDF	87.131	94.535	81.598	91.626	88.722
CurrSDF	87.365	94.611	82.187	92.399	89.141
Ours-FE	88.114	94.635	82.121	93.911	89.695
Ours-PE	87.210	95.722	83.921	93.744	90.149
Ours-LW	87.026	94.602	84.014	92.305	89.457
Ours	**88.137**	**95.910**	**84.135**	**94.215**	**90.599**
CD↓					
模型	Plane	Sofa	Lamp	Shoe	Mean
DeepSDF	0.606	0.770	1.596	0.840	0.953
CurrSDF	0.609	0.764	1.800	0.803	0.994
Ours-FE	0.591	0.758	1.558	0.725	0.908
Ours-PE	0.588	0.748	1.587	0.763	0.922
Ours-LW	0.611	0.762	1.511	0.755	0.910
Ours	**0.582**	**0.734**	**1.472**	**0.703**	**0.873**
F-Score↑					
模型	Plane	Sofa	Lamp	Shoe	Mean
DeepSDF	98.519	97.392	89.356	96.484	95.438
CurrSDF	98.771	97.535	89.915	97.387	95.902
Ours-FE	**99.003**	97.774	90.754	98.725	96.564
Ours-PE	98.021	97.732	91.216	98.068	96.260
Ours-LW	98.525	97.521	91.514	97.217	96.194
Ours	98.921	**98.061**	**93.208**	**99.256**	**97.362**

定量结果表明，改进模型所有的消融版本均能显著提高重建结果的误差指标，而其完整版本在所有类别的测试中均获得了最好的成绩。经过对比不难看出，三个消融版本对不同类别的测试数据的提升效果不尽相同。自适应损失加权策略对含有大量薄壁、细杆结构的测试数据的提升较为明显，如灯类。邻域表面特征编码对表面纹理更复杂和复杂的测试数据更为有效，如鞋类。而阶段性位置编码对所有类别的测试数据上均有稳定的效果。

为了更直观的展示改进模型的性能优势，对所有类别的测试结果进行了可视化，如图 3.6 所示。DeepSDF 无法精确恢复精细的表面纹理以及某些复杂的拓扑结构；CurrSDF 更偏向于保持正确的拓扑结构，虽然提高了重建结果的视觉效果，但会在某些情况下增加表面拟合的平均距离误差；相较前两种方法，改进模型能够获取更精细的纹理细节和更正确的拓扑结构，其重建结果也与真实模型最接近。

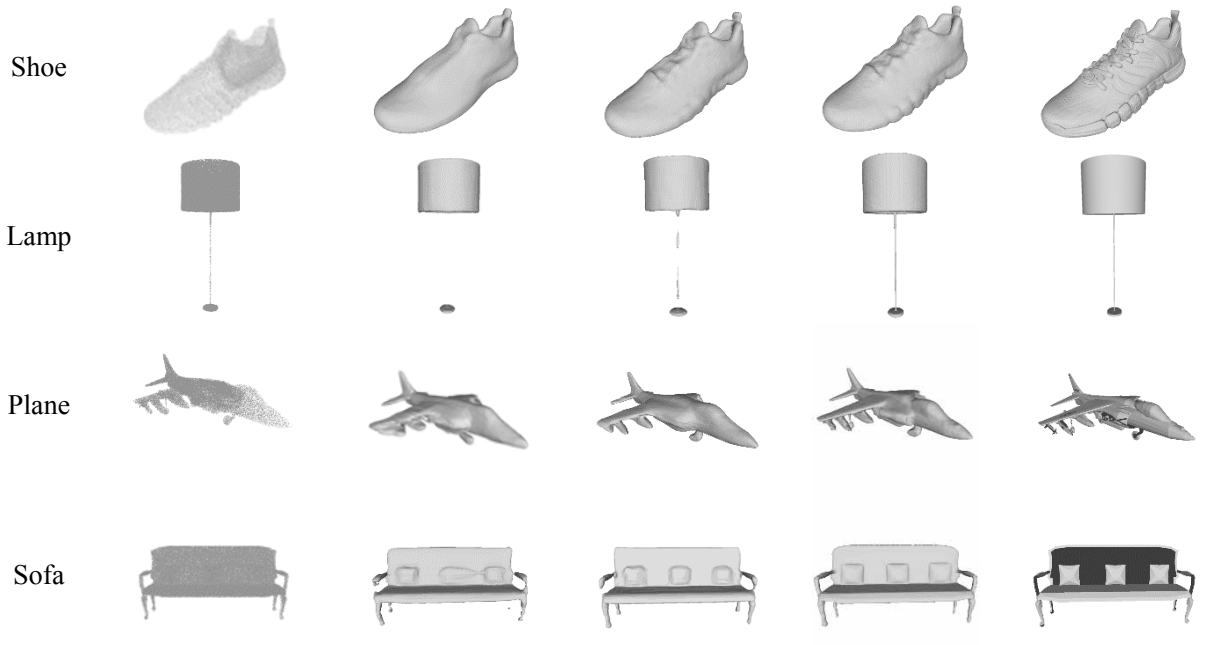


图 3.6 形状重建的定性结果

Figure 3.6 Qualitative results of shape reconstruction

通过定性对比，不难发现改进后的模型重建结果具有明显优势。具体表现在以下几个方面：鞋面和鞋帮的纹理细节被清晰地捕捉；台灯细长的立柱没有破碎或断裂；战斗机两翼下方的起落架没有缺失，机身整体的流线型轮廓更加逼真；沙发扶手更为完整，椅背的形状被正确恢复，三个抱枕的形状更加统一和规则。

图 3.7 展示了改进模型在鞋类数据上进行训练的损失收敛曲线。从整体上看，损失收敛的过程非常的平滑。学习率每 500 个训练周期减半，与之对应损失值也发生了明显

的下降。从损失下降的趋势中可以看出，模型在前 1000 个周期的训练中已经基本掌握了3D对象的整体轮廓，并在随后1000个训练周期中逐渐开始学习更为复杂的表面细节。

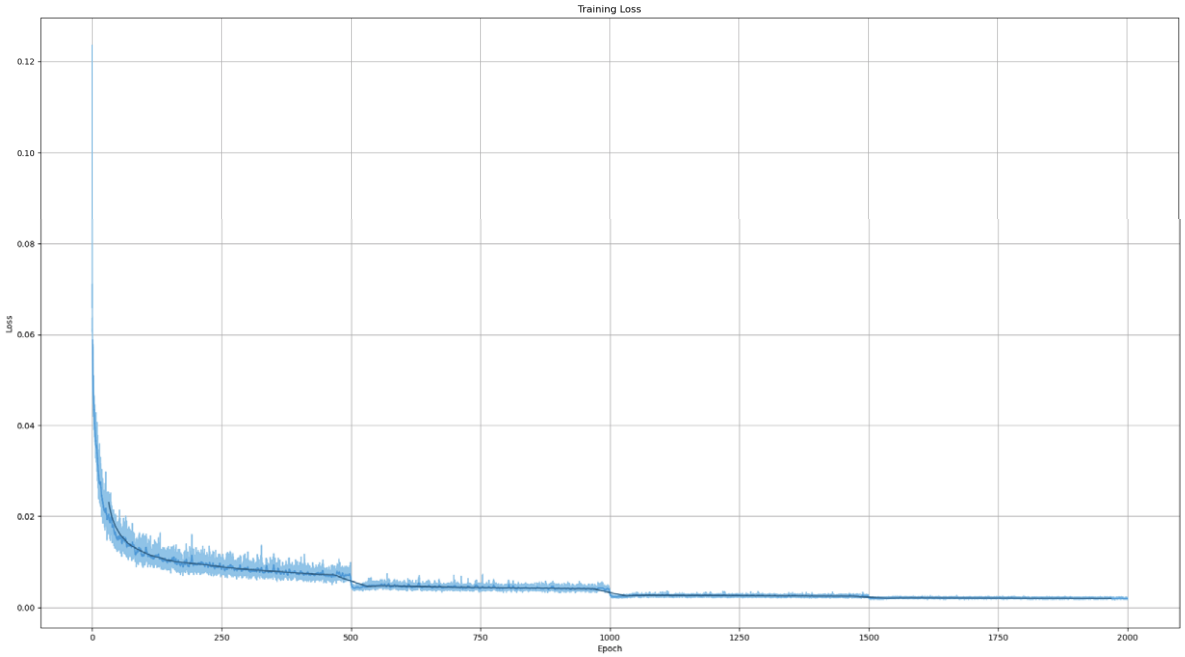


图 3.7 在鞋类数据上的损失收敛曲线

Figure 3.7 Convergence curve of the loss on the shoe data

3.6.2 超参数的合理性研究

为了探索超参数设置对改进模型的性能影响，在鞋类数据上对阶段性位置编码的频率基数 L 和邻域表面特征编码的邻居点数 K 取值的合理性进行了验证，如表 3.2 所示。

表 3.2 超参数设置的合理性验证

Table 3.2 Rationality verification of hyperparameter settings

CD↓

类别	L				K		
	4	6	8	10	4	8	16
Shoe	0.738	0.725	**0.703**	0.704	0.716	**0.703**	0.801

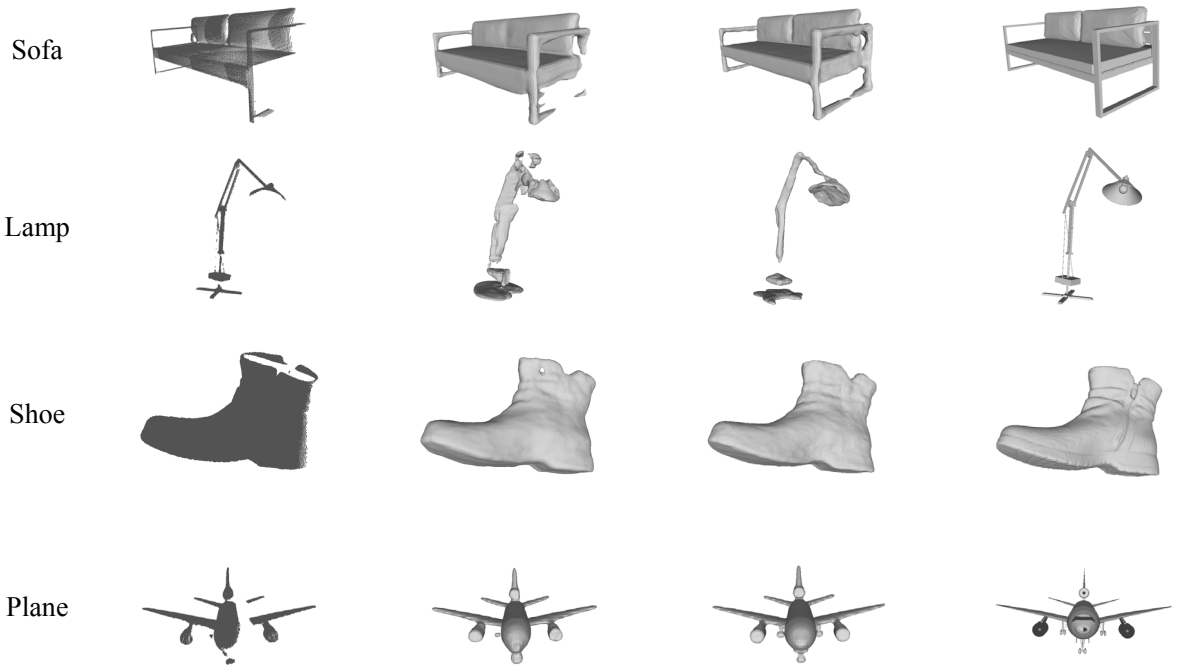
当 L 从 0 增长到 8 时，由于获得了更为丰富的高频信息，模型性能得到不断地提升。当其超过 8 以后，模型性能几乎不再提升。据推测，对采样点坐标进行高频映射的作用是有限的，当其维度超过了形状编码的维度（256）以后，模型对高频细节的学习能力将趋于饱和。因此建议将 L 设置为 8。

当 K 被设置为 4 时，对模型性能的提升作用非常有限。当其被设置为 16 时，模型的性能出现了明显下降。据推测，当邻居点过少时，特征编码难以从邻居点中提取精确

的邻域表面信息。当其过多时，特征编码对邻域表面的几何特征进行了过多的平滑近似，反而增大了表面拟合的偏差。因此建议将 K 设置为 8。

3.6.3 单视角下的形状补全

基于 Auto-Decoder 架构的隐式神经表示方法拥有强大的泛化能力，能够在只有部分观测数据的情况下，仅通过之前学习到的形状先验对当前 3D 对象的其他未知区域进行形状补全。因此，对改进模型与 DeepSDF 在以单视角深度图为输入的形状补全应用中的表现进行了比较，定性和定量结果分别如图 3.8 和表 3.3 所示。



(a) Point Cloud (b) DeepSDF (c) Ours (d) Ground Truth

图 3.8 单视角下的形状补全应用的定性结果

Figure 3.8 Qualitative results of shape completion application in a single view

表 3.3 单视角下的形状补全应用的定量结果

Table 3.3 Quantitative results of shape completion application in single view

类别	DeepSDF			Ours		
	IoU↑	CD↓	F-Score↑	IoU↑	CD↓	F-Score↑
Shoe	77.233	1.865	74.811	86.118	1.541	76.455
Plane	67.971	1.539	81.277	73.536	1.015	89.755
Sofa	79.555	2.312	72.090	81.721	1.912	75.922
Lamp	53.232	3.457	59.417	62.120	2.812	65.175

结果证明，改进模型有更强的泛化能力和鲁棒性，其重建结果精度更高，且能够正确的恢复未知的表面区域。具体来说，在沙发右侧的腿部区域、台灯的连杆和灯罩、靴子鞋口附近的孔洞以及飞机右侧的起落架等位置。

3.6.4 形状编辑

基于 Auto-Decoder 架构的隐式神经表示的另一个重要应用是形状编辑。由于形状编码本质上是 3D 对象的表面控制参数在低维潜空间下的嵌入向量，定向的编辑形状编码即可实现对 3D 对象的表面形状的控制。因此，在鞋类数据上对改进模型在形状编辑应用下的表现进行了可视化验证，如图 3.9 所示。

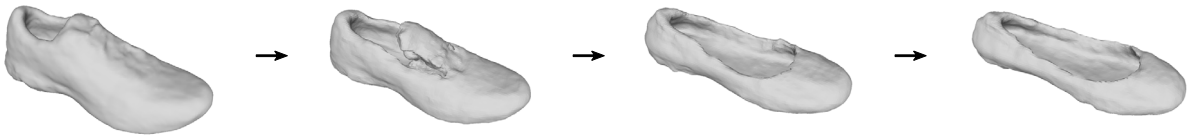


图 3.9 形状编辑应用的展示结果

Figure 3.9 Display results of the shape editing application

这两个中间状态是通过线性插值获得的，它们包含了正确的鞋类外观，几乎没有出现任何破碎或畸形。这证明了改进后的模型具有强大的可编辑能力，能够被广泛的应用于基于形状编辑的下游任务。

3.7 本章小结

本章提出的改进模型采用了编码－解码的架构，显著提高了表面重建的精度。编码器分别对了采样点的邻域表面特征和高维映射进行编码，解码器使用编码信息拟合表面。此外，提出了自适应损失加权策略，加强了解码器对难以表示的局部细节的惩罚。本文分别从 ShapeNet 和 GSO 两个开源数据集上选取了飞机、灯、沙发和鞋四类数据，进行了形状重建、单视角下的形状补全和形状编码任务的实验验证。改进模型在所有实验中均取得了最好的指标成绩和视觉效果。

4 基于图像的隐式神经表面重建的改进

NeRF 是一种基于图像的隐式神经表示，使用 MLP 作为其网络框架。众所周知，MLP 更偏向于学习输入信号中的低频信息，而其捕获高频信息的能力具有上限，该上限与网络参数的范数呈正相关。为解决基于 MLP 的网络框架的局限性，NeRF 提出了位置编码。通过使用连续傅里叶变换，将输入的三维坐标分解成不同的频率基，以更好地近似场景中的高频细节。尽管位置编码可以增强网络捕获高频细节的能力，但也会增加网络过拟合的风险。因为高维映射可能会产生高频噪声，影响网络的鲁棒性和泛化能力。因此，在渲染图像中会出现意外的错误和缺失结构，凸显了将所有高频信号纳入模型的问题。因而得出结论，并非总是有益于包含所有高频信号，因为这可能会使拟合目标函数更加困难，并对网络的性能产生负面影响。

在过去的五年里，注意力机制被广泛应用于人工智能中各种数据驱动的任务，包括自然语言处理（NLP）、计算机视觉（CV）等。注意力机制是一个广义的概念，独立于任何特定的框架。带有注意力机制的框架可以权衡输入信号，学习如何分配自己的注意力，加强输入信号的一些关键特征对后续任务的影响。由于大量基于注意力的模型被提出并取得了优异的性能，注意力机制已经发展成为深度学习中最著名的中心思想之一。其中最著名的框架之一就是 self-attention[47]。然而，一些研究者通过实验测试发现，它需要大量的计算资源并且具有坚实的黑盒特性[48]。最近，一些研究人员试图对前馈神经网络（feedforward neural network，FFN）进行静态参数化，实现了类似的注意力机制[49-51]，并在更高的计算效率下获得了匹配 ResNet[52]和 VIT[53]的图像分类性能。这类基于 MLP 的框架，更为简单和高效，能够有效地应用在基于图片的隐式神经表示中。

在这项研究中，引入了一种新的归纳偏差，通过引入基于注意力的门控单元来分离通道，以加权输入信号中的不同频率分量。与 MLP 架构不同，新的网络架构由多个堆叠的注意力模块组成，通过将静态的注意力分配给不同的输入信号，提高了重建精度。此外，使用球谐函数来显式地预测在不同视角下采样点的颜色值，从而增强了模型在非朗伯场景中的鲁棒性和精度。为了解决新架构增加了计算负担的问题，设计了一种体素结构，通过中缓存场景中采样点 SDF 值的分布，显著提高了采样效率。

此外，上述的改进可以灵活地应用于各种基于 NeRF 的框架，使其在 3D 视觉领域具有重要的价值。这些方法可以提高渲染精度和效率，从而为更多的应用场景提供了可能性，例如计算机视觉、虚拟现实和增强现实等。因此，这些改进点为 NeRF 技术的进一步发展和应用提供了有力支持。

4.1 改进模型

改进模型基于 Neus[21]，这是一种优秀的基于图片的隐式神经表面重建方法。Neus 简洁的架构和出色的性能，能够轻松地对其实施改进并验证其有效性。改进模型的输入和输出协议也与 Neus 保持一致，以保证其兼容性和可靠性。

改进模型使用了一种连续的 5D 映射函数来对其进行表示，这类似于原始的 Neus。这种表示避免了使用体素网格来显式地记录 3D 对象的属性值，从而实现了更高效的内存使用。具体而言，参数化了整个 3D 对象表面的神经网络可以从一对 3D 坐标 $p=[x,y,z]$ 和观察方向 $d\in\mathbb{R}^3$ 中推断出相应的颜色和 SDF 值，其表达式为：

$$(c,s)=f(p,d) \tag{4.1}$$

式中：$c=(r,g,b)$ 是一个以依赖观测方向的颜色；s 表示 p 所在位置的 SDF 值。

3D 对象表面可以使用 SDF 的零水平集来表示，即：

$$S=\left\{x\in\mathbb{R}^3 \mid f(x)=0\right\} \tag{4.2}$$

使用 MC 算法，即可从训练完成的网络中显式提取出完整的网格模型。

4.1.1 颜色场与 SDF 场

改进模型通过定义 SDF 场和颜色场，来精确的表征 3D 场景中的 SDF 值和颜色值分布，其表达式为：

$$\begin{aligned} D_\theta &: p\rightarrow(s,f) \\ C_\theta &: (f,d,p,\nabla s)\rightarrow c \end{aligned} \tag{4.3}$$

式中：D_θ 表示 SDF 场的映射函数，它将空间位置 p 映射为她到 3D 对象表面的 SDF 值；C_θ 表示颜色场的映射函数，它对颜色进行编码，将点 p 与观察方向 d 相关联；θ 表示映射函数的参数；f 表示与颜色相关的一个特征向量；∇s 表示 SDF 值的法向量方向。

在 Neus 中，SDF 场和颜色场均是由两个基础的 MLP 表示的，如图 4.1 所示。SDF 场的网络包括八个全连接层，每层有 256 个通道，并使用 Softplus[54]作为激活函数（$\beta=100$）。在第五层，它使用了一个跳跃连接，将输入重新接入到网络中。其输出结果为 $s\in\mathbb{R}$ 和 $f\in\mathbb{R}^{256}$；颜色场网络包括四个全连接层，每层有 256 个通道，并使用 ReLU[55]激活函数。其输出结果为 $c\in\mathbb{R}^3$。在训练时，这些 MLP 的参数会被学习，以便最好地对场景的 SDF 值和颜色值的分布进行建模。

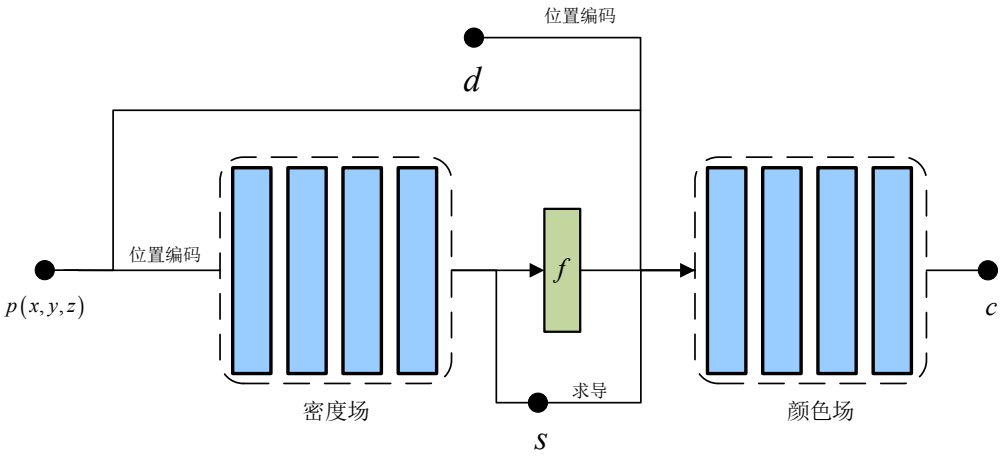


图 4.1 Neus 的原始架构

Figure 4.1 The original architecture of Neus

改进的模型引入了注意力机制，通过使用注意力模块来突出输入信号中的高频特征，并抑制其中的高频噪声。每个注意力块相当于两个全连接层组成的 FFN。对于 SDF 场，使用四个串联的注意力模块替换原始的 MLP 网络。每个注意力模块的结构和尺寸完全相同，其输入和输出的维度均为 256，与原始的 SDF 场一致。对于颜色场，引入了球谐函数来显式地定义光线的球面反射情况，以提高体积渲染过程的前向传播效率。它使用神经网络来预测一组球谐系数，并通过计算球谐函数，输出在特定视角下的颜色观测值。该神经网络由一个全连接层和一个注意力模块组成，其输入和输出的维度均为 256，与原始颜色场网络的设置一致。为了使注意力模块满足模型输入和输出的维度要求，通常会在注意力模块前面添加一个全连接层。这个全连接层通常不使用任何激活函数，只是简单地执行维度的匹配。总而言之，注意力模块能够增强模型捕获输入信号中复杂特征的能力，从而提高了模型性能。

4.1.2 体积渲染

改进模型使用了与 Neus 相同的体积渲染方法，仅以带有相机参数的 2D 输入图像作为监督信号来训练网络。在使用这种监督信号成功地最小化损失函数后，网络编码的 SDF 的零级集能够准确地表示重建表面 S 。

空间中的任何光线都有一定的概率穿过客观场景并被其介质发生相互作用，因此，特定位置的密度值和颜色值由穿过它的光线共同决定。体积渲染采用类似蒙特卡罗的采样策略来离散化连续场景，该场景是可微的并且在反向传播期间提供足够的梯度，其计算方法如式（2.8）所示。

Neus 使用了标准的逻辑斯谛密度函数来定义 SDF 场的密度分布，称为 S-密度，其表达式为：

$$\sigma_s(s)=\frac{1}{\gamma}\frac{\exp\left(\frac{s}{\gamma}\right)}{\left(1+\exp\left(\frac{s}{\gamma}\right)\right)^2} \tag{4.4}$$

式中：s 为 SDF 值；γ 为 S-密度的标准差，描述了其散布程度；$\sigma_s(\cdot)$ 为 Sigmoid 函数的导数。

给定一条从相机光心位置 o 朝 v 方向发射的光线，经过一系列的反射、折射和散射之后，最终抵达图像平面上对应的像素点位置。该像素点的预测值 $\hat{C}(o,v)$ 可以通过对光线经过路径上所有位置的密度和颜色值进行积分获得，其表达式为：

$$\hat{C}(o,v)=\int_0^{+\infty}\omega(t)c\left(p(t),v\right)dt \tag{4.5}$$

式中：$p(t)=o+tv$ 表示光路上的一个点，$t\geq 0$ 表示光线传播的距离；$\omega(\cdot)$ 表示权重函数，$c(\cdot)$ 表示颜色函数。

从 2D 图像中学习准确的 SDF 场的关键在于建立输出颜色和 SDF 之间的适当联系。具体来说，需要根据场景的 SDF 场分布，得到光线路径上每个位置适当的权重函数。然后，使用式（4.5）计算该光线在图像上投影位置的像素值。最后，通过最小化该值与对应监督值之间的损失，来反向优化 SDF 场的分布。

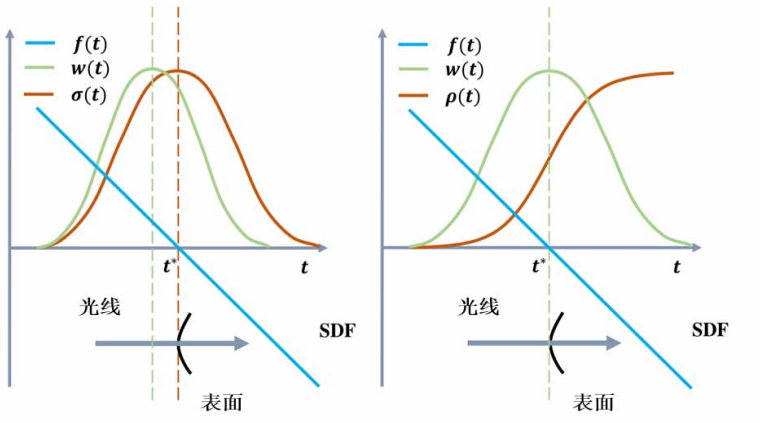
为了实现准确的体积渲染，权重函数需要满足两个关键特征。首先，它必须是无偏的，即渲染过程中不应引入任何系统误差或偏差，以确保输出结果与实际场景一致。具体而言，当光线与 3D 对象表面接触时（即 SDF 值为零），权重函数应达到其局部最大值，以确保符号距离函数（SDF）的零水平集准确表示 3D 物体的几何形状。其次，权重函数应该具有遮挡感知的特性，即随着光线传播，权重值应整体下降，以避免输出结果中出现不正确的遮挡关系。当场景中存在多个或复杂的 3D 对象时，光线可能会多次穿入和穿出表面。当光线上的多个采样点具有相同的 SDF 值时，它们似乎对最终输出颜色的贡献相同，因为它们与表面的距离相等。但是，如果其中一个点的前方存在其他的点遮挡，则它不应该对最终输出颜色起到关键作用。权重函数的遮挡感知确保了更靠近相机的点将被分配更高的权重。

正如 2.2.5 节中所介绍的，标准的体积渲染公式中的权重函数具有遮挡感知的能力，其表达式为：

$$\begin{aligned}\omega(t)&=T(t)\rho(t)\\ T(t)&=\exp\left(-\int_0^t\rho(u)du\right)\end{aligned} \tag{4.6}$$

式中：$\rho(t)$表示不透明度；$T(t)$表示透光率。

如果直接使用 S-密度对体积密度进行替换，会引入一个固有的偏差，使得权重函数在光线接触表面之前的某个位置就已经达到了局部最大值，如图 4.2（a）所示。



（a）有偏权重　　（b）无偏权重

图 4.2 权重函数的偏差分析

Figure 4.2 Bias analysis of weight function

首先，假设只考虑光线单次穿过 3D 对象表面的简化情况，即只有一个表面交点。在此情况下，对 S-密度进行归一化直接将其构造为权重函数，可以满足无偏性要求，其表达式为：

$$\omega(t)=\frac{\sigma_s\left(f\left(p(t)\right)\right)}{\int_0^{+\infty}\sigma_s\left(f\left(p(u)\right)\right)du} \tag{4.7}$$

在这种情况下，一条光线上任意位置的 SDF 值可以表示为：

$$f\left(p(t)\right)=\left|\cos\theta\right|\cdot(t_s-t) \tag{4.8}$$

式中：t_s表示光线与表面的接触点，即$f\left(p(t_s)\right)=0$；θ表示表面点t_s的法线方向与光线传播方向的夹角，并且在同一条光线上处处相等。

因此，式（4.7）中的权重函数可以被简化为：

$$\omega(t)=\left|\cos\theta\right|\sigma_s\left(f\left(p(t)\right)\right) \tag{4.9}$$

然而，这个权重函数并不具备遮挡感知的能力。当光线多次穿过 3D 对象的表面时，光路上会存在多个 SDF 值为 0 的点。虽然该函数能够准确地预测权重峰值的位置，但它无法判断这些位置之间的遮挡关系，因此会导致渲染结果的错误。

为了同时满足遮挡感知和无偏的特性，需要将 S-密度转化为标准体积渲染中不透明度$\rho(\cdot)$的对应形式，即：

$$T(t)\sigma(t)=\left|\cos\theta\right|\sigma_s\left(f\left(p(t)\right)\right) \tag{4.10}$$

根据式（4.6）可以轻松的证明$T(t)\sigma(t)=-\frac{\mathrm{dT}}{dt}(t)$。

定义一个与 S-密度相关的函数$\Phi_s(\cdot)$，其满足以下关系：

$$\begin{aligned}\frac{d\Phi_s}{dt}\left(f\left(p(t)\right)\right)&=\frac{\mathrm{dT}}{dt}(t)\\&=-\left|\cos\theta\right|\sigma_s\left(f\left(p(t)\right)\right)\end{aligned} \tag{4.11}$$

在简化情况下，通过结合式（4.6）和式（4.11）进行计算，可以得出 S-密度在体积渲染公式中的不透明度形式，其表达式为：

$$\rho_s(t)=-\frac{\frac{d\Phi_s}{dt}\left(f\left(p(t)\right)\right)}{\Phi_s\left(f\left(p(t)\right)\right)} \tag{4.12}$$

此时，考虑将$\rho_s(t)$推广到光线多次穿过 3D 对象表面的一般情况，即存在多个表面交点。当光线到达 3D 对象内部的中心点时，其 SDF 值会取到局部最小值。随后光线将穿出 3D 对象的表面，这个阶段的 SDF 值会逐渐增加，导致$\rho_s(t)$变为负值。因此，为确保不透明度始终为非负值，需要将该阶段的$\rho_s(t)$截断为零，以满足体积渲染公式的定义。其表达式为：

$$\begin{aligned}\omega_s(t)&=T(t)\rho_s(t)\\\rho_s(t)&=\max\left(-\frac{\frac{d\Phi_s}{dt}\left(f\left(p(t)\right)\right)}{\Phi_s\left(f\left(p(t)\right)\right)},0\right)\end{aligned} \tag{4.13}$$

新的 SDF 权重函数$\omega_s(t)$，同时具备了遮挡感知和无偏的特性，分别如图 4.2（b）和图 4.3 所示。

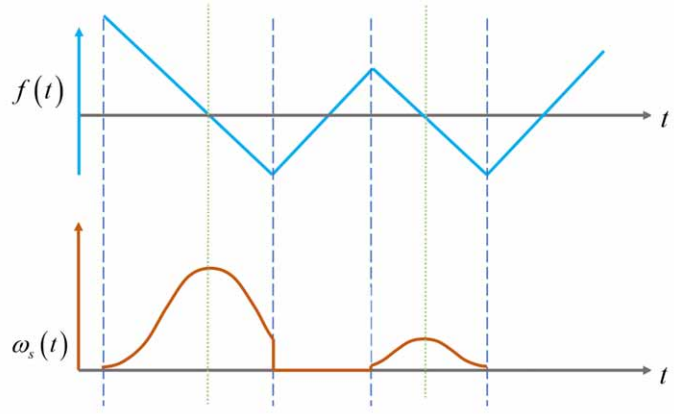


图 4.3 权重函数与 SDF 的对应关系

Figure 4.3 Correspondence between weight function and SDF

4.1.3 离散化采样

在实际的体积渲染计算过程中，为了平衡精度和效率，通常会对连续的场景进行抽样。具体而言，在一条光线上以一定得间隔采样一组点，以近似估计相应像素点的颜色值，其计算公式类似于式（2.10）。这种离散化过程能够有效地减少计算量，提高渲染速度，同时还能够保证渲染结果的质量。

NeuS 采用了分层采样的方法来进行渲染。这种方法分为粗略采样和精细采样两个阶段。在粗采样阶段，首先沿着光线均匀采集 64 个点。接着，使用神经网络预测每个点的 SDF 值。在精细采样阶段，采用四次连续的迭代过程。在每次迭代中，首先使用一个固定的 S-密度标准差计算光线基于粗采样的概率密度函数（Probability Density Function，PDF），并将其转换为累积分布函数（Cumulative Distribution Function，CDF）。然后，在 CDF 上等间隔地采集 16 个点，并使用神经网络预测其 SDF 值。采样完成后，使用一个可学习的 S-密度标准差，预测每条光线上 128 个采样点的 S-密度，并在反向传播时对其进行优化，以输出体积渲染最终的预测结果。

粗采样在训练的初期起着关键作用，因为它对光线经过空间场景的全过程进行了采样。这使得模型可以估计 3D 对象的大致占用情况，并引导更精细的重要性采样。这有效地避免了 3D 对象的拓扑错误，提高了渲染的准确性和效率。然而，随着训练的进行，模型变得更加有经验。如图 4.3 所示，只有位于表面附近的采样点才有较高的权重，而这些采样点只占所有粗采样点中的极小一部分，大约为 10%~20%。相反，大部分粗采样点都处于表面真实位置的拉依达准则之外，对体积渲染的贡献很小，这就导致了计算资源的极大浪费。

由于每个点的 SDF 值由所有经过该点的光线共享，因此可以通过记住全局 SDF 值的分布，来避免对场景中的空白区域进行采样。因此，提出了有效样本的概念，通过使用了一种体素网格来对场景进行粗略的划分。在训练过程中，每个子体素的平均 SDF 值被缓存并动态更新，以便对采样点的有效性进行估计。通过动态剪枝来规避网络对无效采样点的前向传播过程，从而提高了训练的效率。

4.1.4 损失函数

改进模型能够在没有任何 3D 监督的情况下，通过最小化渲染图像与真实监督之间的误差来拟合 3D 对象的表面。其损失函数的表达式为：

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \lambda \mathcal{L}_{\text{reg}} \tag{4.14}$$

式中：$\mathcal{L}_{\text{color}}$ 和 $\mathcal{L}_{\text{reg}}$ 分别表示颜色损失和几何正则化损失。

颜色损失的表达式为：

$$\mathcal{L}_{\text{color}}=\sum_{r\in\mathcal{R}}\left\|\hat{C}(r)-C(r)\right\|_2 \tag{4.15}$$

式中：$\mathcal{R}$ 表示训练集中一批光线；$C(r)$ 表示真实监督值。使用 L1 损失对异常值具有鲁棒性，能够保证训练的稳定。

几何正则化损失，其表达式为：

$$\mathcal{L}_{reg}=\sum_{r\in\mathcal{R}}\sum_{i}\left(\left\|\nabla f\left(p_{r,i}\right)\right\|_2-1\right)^2 \tag{4.16}$$

式中：$\nabla f(\cdot)$ 表示 Eikonal 项；$i\in(0,128)$ 表示一条光线上的采样点索引。

4.2 静态注意力模块

静态注意力模块通过跨通道的静态投影[51]，将注意力分配给不同的频率的输入信号，可以有效地屏蔽输入信号的高频噪声，如图 4.4 所示。其中：□ 表示向量之间的逐元素相乘；⊕ 表示恒等映射操作，用于残差结构[52]中。

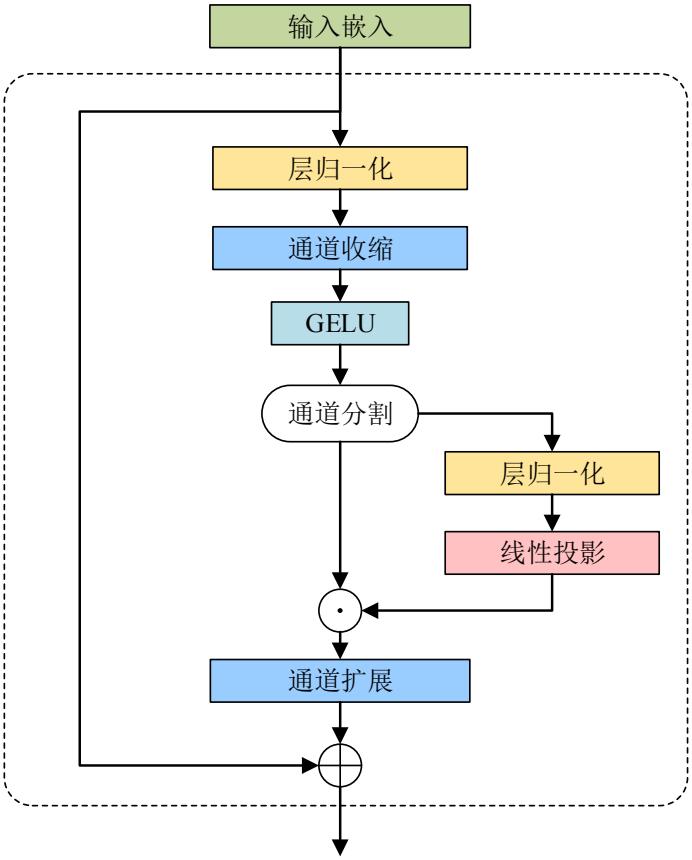


图 4.4 注意力模块的架构细节

Figure 4.4 Architectural details of the attention block

注意力模块专注于坐标位置和观测方向的高频嵌入向量，因此首先使用傅里叶变换

对输入 3D 坐标值进行高频编码。其表达式如式 3.3 所示。注意力模块由三个环节组成：通道扩展、通道收缩和通道的线性投影，其表达式为：

$$\begin{aligned} y &= f_{\text{exp}}(x) \\ z &= s(y) \\ \tilde{x} &= f_{\text{con}}(z) + x \end{aligned} \tag{4.17}$$

式中：$x \in \square^{n}$ 和 $\tilde{x} \in \square^{n}$ 分别表示注意力模块的输入和输出；$y \in \square^{4n}$ 和 $z \in \square^{2n}$ 分别表示模块内的中间变量；$f_{\text{ext}}(\cdot)$ 和 $f_{\text{con}}(\cdot)$ 分别表示通道的扩展和收缩；$s(\cdot)$ 表示通道的线性投影。

通道扩展是一种能够增加特征表示能力的方法，它可以将来自不同频率基上的特征分布更为稀疏，从而有利于后续的操作对其进行更加精准的识别和过滤。具体来说，它有由一个归一化层、一个全连接层和一个 GELU[56]激活层组成。通道的线性投影由一个基于注意力的门控单元（attention-based gating unit，AGU）实现。AGU 能够通过分配不同坐标在不同频率基上的高频分量的注意力，实现对通道的线性投影进行调节，是注意力模块中的核心环节。通道收缩过滤掉了无贡献的特征通道，保证了注意力模块的输入和输出维度的一致性，从而降低了该模块的串联难度。具体来说，它由一个全连接层和一个恒等映射组成。Neus 通过额外使用一个跳过连接操作来提高网络的深度，但其作用有限。恒等映射将输入和输出的响应进行叠加，彻底避免了梯度消失的问题。因此，使用注意力模块的架构在理论上可以比原始的 MLP 架构更深。

4.2.1 AGU

AGU 通过分配通道注意力来改善来自不同频率基上的信号之间的信息交互，它由三个操作组成。首先，将输入 y 沿通道维度均分为两个相同的子向量 $y_1 \in \square^{2n}$ 和 $y_1 \in \square^{2n}$。其次，对其中的一个子向量进行线性变换操作，得到通道注意力权重。最后，将两个子张量逐元素相乘以分配该通道注意力。其完整流程的表达式为：

$$\begin{aligned} [y_1, y_2] &= f_{\text{split}}(y) \\ f_{W,b}(y_2) &= W y_2 + b \\ s(y) &= y_1 \square f_{W,b}(y_2) \end{aligned} \tag{4.18}$$

式中：$f_{\text{split}}(\cdot)$ 表示通道分割操作；$f_{W,b}(\cdot)$ 表示线性变换函数；$W \in \square^{2n \times 2n}$ 和 $b \in \square^{2n}$ 分别表示线性变换的权重矩阵和偏置向量；$\square$ 表示逐元素相乘操作。为了表示的简洁，归一化操作被省略。

对于权重矩阵 W 和偏置向量 b 来说，它们记录了静态参数化的通道注意力。在训练的初期，它们被初始化为 1 和 0，这意味着它们并不会发挥任何作用。然而，随着训练

的进行，通道注意力逐渐被注入到这些参数中，从而引导网络对输入数据进行更好的表示学习。因此，这些静态参数化的通道注意力也可以被视为一种重要的正则化机制，能够帮助网络更好地适应训练数据并提高其泛化能力。

4.3 球谐函数

自然界中的大多数场景都无法符合朗伯模型，因为光线在非朗伯表面上的反射呈现出不同程度的漫反射。漫反射的程度很大程度上取决于表面的粗糙程度。越粗糙的表面越接近朗伯表面，而越光滑的表面越接近镜面。因此，在不同视角下观察同一位置的场景时，可能会出现完全不同的颜色，这与光度一致性准则相矛盾。为了解决这个问题，NeRF 使用神经网络来学习这种复杂的光线变化，并预测特定观测视角下的颜色值。

球谐函数[57]是一种用于模拟单位球上环境光照的函数，最早由数学家 Legendre 在 18 世纪提出。它将周围环境中的光照信息投影到一个球面上，通过将一个复杂的球形函数分解成一系列简单的球谐函数的加权和，来对其进行低维表示，从而更容易地进行分析和计算。这种表示方法可以使渲染引擎在计算光照时更加高效，并且可以轻松地实现漫反射、镜面反射和折射等光学效果。球谐函数具有许多非常优秀的性质，比如正交性、归一性，可递推性和旋转不变性等，已经被广泛应用于模拟各种表面，包括朗伯表面和光滑表面。

球谐函数是定义在单位球面上的复函数，即 $f_{\mathrm{sh}}:\mathbb{S}^2\to\mathbb{C}$，其表达式为：

$$f_{\mathrm{sh}}=\sum_{l=0}^{\infty}\sum_{m=-l}^{l}k_l^m Y_l^m\left(\theta,\phi\right) \tag{4.19}$$

式中：m 表示角量子数，它决定了球面上的角分辨率；l 表示磁量子数，它决定了球面上的方向；$k_l^m\in\mathbb{C}^3$ 表示一个球谐系数，它是一种广义的傅里叶系数，与颜色相关；$Y_l^m\left(\cdot\right)$ 表示球谐函数中的一个归一化基函数，其表达式为：

$$Y_l^m(\theta,\phi)=(-1)^m\sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos\theta)e^{im\phi} \tag{4.20}$$

式中：$0\le l\le l_{\max}$ 和 $-l\le m\le l$ 分别表示两个不同的控制参数；ϕ 和 θ 分别表示球面上的极角和方向角；$P_l^m(\cdot)$ 表示缔结勒让德多项式，它描述了球面上的角度依赖性，其表达式为：

$$\begin{aligned}P_l^m(x)&=(-1)^m(1-x^2)^{m/2}\frac{d^m}{dx^m}P_l(x)\\P_l(x)&=\frac{1}{2^l l!}\frac{d^l}{dx^l}(x^2-1)^l\end{aligned} \tag{4.21}$$

式中：$P_l^m(x)$表示连带勒让德多项式，它描述了球面上的角度无关性。

为了减少网络参数量，采用了 Yu 等人的方法[58]，在体积渲染阶段中应用球谐函数。具体来说，改进模型调整了颜色场网络的架构，以输出一组球谐系数k，而不是直接输出 RGB 值，其表达式为：

$$C_\theta:(f,p,\nabla s)\to\left\{k_l^m\right\} \tag{4.22}$$

随后，坐标位置p在某个特定观测方向下的颜色值c可以通过查询球谐函数来确定，其表达式为：

$$c=f_{\mathrm{sh}}(k,d) \tag{4.23}$$

换言之，使用球谐函数能够解耦颜色与观测方向之间的联系，避免了需要使用神经网络对观测视角进行建模，从而消除了在渲染阶段重新进行采样的过程。这样，可以直接查询任意角度下的颜色值，从而提高了模型的训练速度。

4.4 采样点剪枝策略

根据公式（4.13），可以发现当一个采样点的 SDF 值越大时，其对于光线的体积渲染结果产生的贡献就越小，因为其不透明度越低。然而，在大部分场景中存在大量无效的空白区域，它们对场景的理解没有任何贡献。这就导致了大量的计算资源被浪费，因为许多光线甚至没有穿过场景中的 3D 对象表面。一旦知道这些无效采样点的位置，就可以在之后的训练过程中将它们从网络中剔除。对采样点进行剪枝，可以大大减少计算量，从而提高渲染效率。

为了实现这一目标，构建了一个 SDF 体素网格$\mathrm{V_s}$，其分辨率为$\mathrm{D\times D\times D}$。在训练过程中，$\mathrm{V_s}$用于缓存目标场景的全局 SDF 值分布，如图 4.5 所示。为了控制合理的采样区间，引入截断控制参数δ，即只有 SDF 的绝对值小于δ的采样点才被视为有效样本。较大的δ允许模型进行更快速的光线追踪，因为每个样本都提供安全步长的信息。较小的δ允许模型将注意力集中在表面附近的细节上。根据经验，将其设置为 0.1。在$\mathrm{V_s}$中，所有子体素的默认 SDF 值均初始化为零，这意味着在训练初期，所有的采样点都是有效样本。

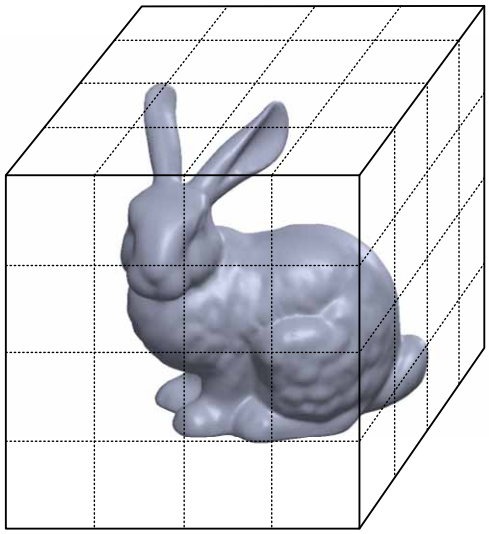


图 4.5 SDF 体素网格的划分方式

Figure 4.5 Schematic diagram of the SDF voxel grid

为了对任意采样点 p 进行 SDF 值的计算，首先需要将其对应到体素网格中的索引位置。其表达式为：

$$i = \frac{p - p_{\min}}{p_{\max} - p_{\min}} \mathrm{D} \tag{4.24}$$

式中：$p_{\min}$ 和 $p_{\max}$ 分别表示空间场景区域的最大和最小边界值。

然后，根据该索引位置查询相应的子体素，使用动量法来更新其 SDF 值。动量更新的表达式如下：

$$V_{\mathrm{s}}[i] \leftarrow (1-\beta) \cdot V_{\mathrm{s}}[i] + \beta \cdot f(p) \tag{4.25}$$

式中：$\beta \in [0,1]$表示更新控制参数。

在训练初期，所有采样点都被视为有效样本，并更新其所在的子体素值。随着训练的进行，动量 SDF 体素网格能够反映整个场景的最新 SDF 分布。此时，可以通过查询采样点的有效性来判断其是否需要进行更新。如果该采样点被视为有效样本，则会将其输入到神经网络中进行预测，并使用预测结果更新子体素。否则，直接将子体素的值作为该采样点的 SDF 值，并将其用于体积渲染，从而节省计算量，增加采样效率。

4.5 实验设置

4.5.1 数据集

DTU MVS 2014（DTU）[59]是一个用于多视图立体重建的室内场景数据集，由丹麦技术大学（Technical University of Denmark）的计算机视觉和图形图像组织创建并维护，被广泛应用于评估机器学习算法在 3D 建模和表面重建任务中。DTU 数据集由 80 个高

精度的室内场景组成，每个子数据集包含使用搭载结构光扫描仪的工业机械臂从 49 或 64 个相机位置拍摄的 RGB 图像，这些图像用于描述物体表面。每张图像分辨率均为 1600 x 1200，并且已经通过 Matlab 校准工具箱获得高精度的相机位置和内部相机参数。与此同时，每个子数据集还包括一个通过高精度光学扫描技术获取的三维模型作为参考，其精度可达到 0.1 毫米。此外，每个子数据集都具有从定向到漫射的七种不同照明条件。

BlendedMVS[60]是一个用于多视角立体重建的大规模数据集，由苏黎世联邦理工学院（ETH Zurich）的计算机视觉与几何组（CVG）创建并维护。该数据集包含 113 个场景，涵盖城市、建筑、雕塑和小物体等多种场景内容。每个场景包含几十至上百张高分辨率图像，这些图像是由多个不同的相机在场景的不同位置和角度拍摄而成，并包含详细的相机参数和姿态信息。相比于 DTU 数据集，BlendedMVS 的图像捕捉风格和轨迹更加多样，包括不同光照条件、移动物体、纹理缺失和重叠等具有挑战性的情况，更贴近真实世界下的应用场景，因此对算法的泛化能力有更高的要求。

Realistic Synthetic 360° 数据集[14]是由谷歌公司推出的由一个 Blender 软件合成的 360 度全景图像数据集。该数据集包含八个复杂的非朗伯场景，每个场景提供了 100 张训练图像和 200 张测试图像，这些图像是由以场景为中心的上半球面上设置的虚拟相机拍摄的。每张图像的分辨率为 800 × 800 像素，并且还提供了相机详细的注释信息，包括相机位置、拍摄方向和场景深度等。

4.5.2 实施细节

为了确保比较的公平性，所有的实验遵循与 Neus 完全一致的训练与测试集划分和训练周期。使用 RAdam 优化器来改进模型进行了 30 万次迭代的训练。在每个优化步骤中，从训练集中连续采样了 512 条光线。3D 坐标 p 和观测方向 d 的位置编码频率基数分别设置为 6 个和 4 个。动量 SDF 体素网格的分辨率 D 被设置为 384，更新控制参数 β 被设置为 0.1。球谐函数的次数设置为 3，即球谐系数 k 的维度为 49。为了保证训练的稳定性，学习率在前 5 千次迭代中从 0 线性预热（warm up）到 5×10^{-4} 来避免早期过拟合，并且此阶段强制将所有采样点视为有效样本。在预热阶段之后，使用余弦函数将学习率逐渐降低到大约 2.5×10^{-5}。训练完成后，使用 MC 算法即可从 SDF 场中提取出完整的网格模型。此外，所有的实验均是在一台配备 Nvidia RTX-2080Ti GPU 的服务器上进行的。

4.6 分析与讨论

4.6.1 形状重建

在 DTU 数据集上，分别使用带有掩码（w/ mask）和不带掩码（w/o mask）的图像对改进模型进行训练，并使用 CD 指标来衡量其重建质量。为了全面评估改进模型的性

能，将实验结果与该领域几项相关工作进行了比较。Neus 是实验的主要基线（baseline），直接从原始论文中汇报了其定量结果。除此之外，还将改进模型与一种经典的传统表面重建方法 COLMAP[61]（修剪值设置为 0），以及一种早期的隐式神经表面重建方法 IDR[18]进行了比较。实验结果的完整数据如表 4.1 所示。

表 4.1 DTU 数据集表面重建的定量结果

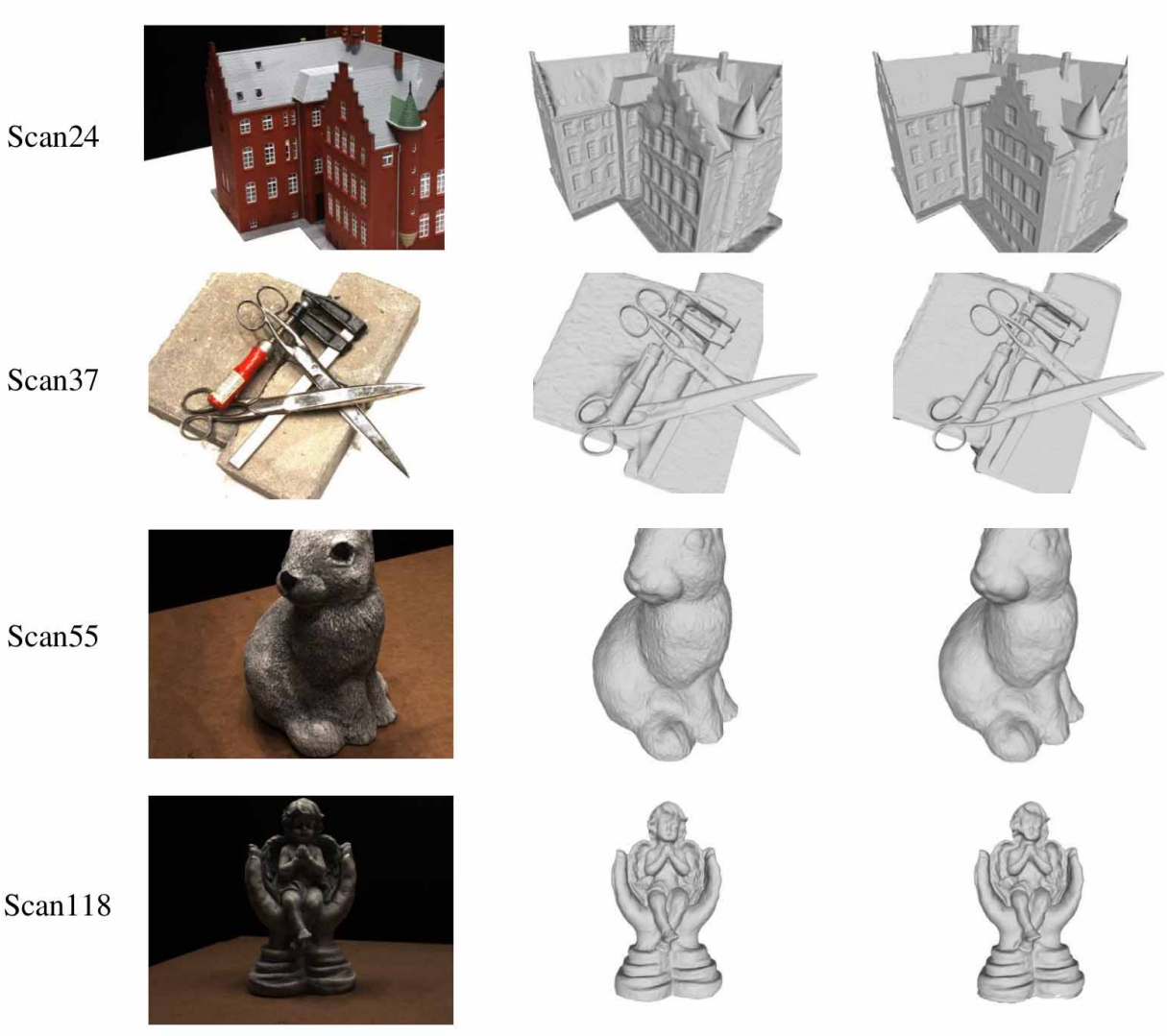
Table 4.1 Quantitative results of surface reconstruction on the DTU dataset

CD↓

ScanID	w/ mask			w/o mask		
	IDR	Neus	Ous	COLMAP	Neus	Ous
Scan 24	1.63	0.83	**0.72**	**0.81**	1.00	0.94
Scan 37	1.87	0.98	**0.85**	2.05	1.37	**1.07**
Scan 40	0.63	0.56	**0.44**	**0.73**	0.93	0.82
Scan 55	0.48	**0.37**	0.38	1.22	**0.43**	0.51
Scan 63	**1.04**	1.13	1.08	1.79	1.10	**1.01**
Scan 65	0.79	0.59	**0.55**	1.58	0.65	**0.58**
Scan 69	0.77	0.60	**0.52**	1.02	0.57	**0.55**
Scan 83	1.33	1.45	**1.02**	3.05	1.48	**1.14**
Scan 97	1.16	0.95	**0.82**	1.40	1.09	**0.98**
Scan 105	0.76	0.78	**0.69**	2.05	**0.83**	0.90
Scan 106	0.67	0.52	**0.48**	1.00	0.53	**0.51**
Scan 110	**0.90**	1.43	1.21	1.32	**1.20**	1.24
Scan 114	0.42	**0.39**	0.41	0.49	**0.35**	0.42
Scan 118	0.51	0.45	**0.40**	0.78	0.49	**0.45**
Scan 122	0.53	**0.45**	0.47	1.17	0.54	**0.51**
Mean	0.90	0.77	**0.67**	1.36	0.84	**0.78**

经过分析，改进的模型在大多数情况下都能够提供更高质量的重建结果。在存在掩码的情况下，改进模型的平均 CD 值比 Neus 低了 0.1。在不存在掩码的情况下，改进模型的平均 CD 值比 Neus 降低了 0.06。

为了更直观地对比改进模型和基线模型在 DTU 数据集和 BlendedMVS 数据集中的重建效果差异，在图 4.6 和图 4.7 中展示了部分场景下的重建结果。其中，图 4.6 展示了使用掩码的情况，而图 4.7 展示了不使用掩码的情况。



（a）Ground Truth　　（b）Neus　　（c）Ours

图 4.6 带掩码监督的表面重建定性比较（DTU）

Figure 4.6 Qualitative comparison of surface reconstruction with mask supervision (DTU)

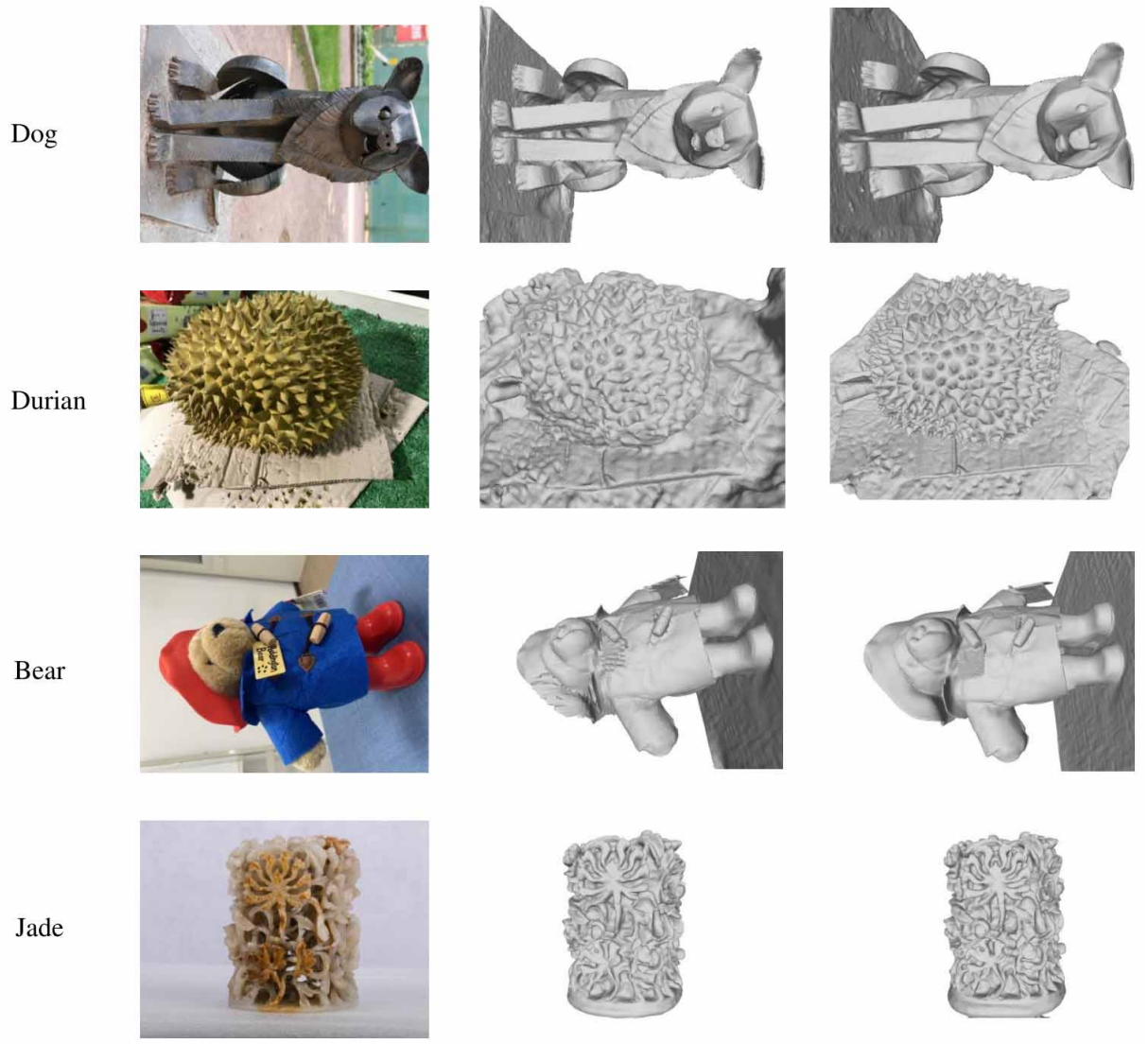


图 4.7 不带掩码监督的表面重建定性比较（BlendedMVS）

Figure 4.7 Qualitative comparison of surface reconstruction without mask supervision (BlendedMVS)

从图中不难发现，相比于基线模型，改进模型能够产生更为光滑、自然、细腻的表面。特别地，在使用掩码的情况下，改进模型在重建表面时能够呈现更为连续、一致的细节，使得边缘处的重建结果更加贴合真实模型。此外，改进模型还能够更好地处理遮挡、透视等复杂场景，使得重建结果更加准确。

4.6.2 消融研究

为了评估改进模型中不同模块对整个系统性能的贡献，使用 DTU 数据集在带掩码的条件下，进行了一系列的消融实验。以 Neus 为基线模型，并在其基础上分别加入了注意力模块、球谐函数和采样点剪枝，以研究它们对模型性能的功效。消融研究的定量

结果如表 4.2 所示。

表 4.2 DTU 数据集消融研究的定量结果

Table 4.2 Quantitative results of ablation study on the DTU dataset

注意力模块	球谐函数	采样点剪枝	CD ↓	训练时间（h） ↓
			0.77	**16**
√			0.68	27
√	√		0.66	24
√	√	√	**0.67**	18

根据表中数据，可以看出使用注意力模块对于提高重建结果的质量非常有效，但是相应地大大增加了训练时间。球谐函数可以改善某些场景下的重建质量，并且通过优化渲染阶段缩短了训练时间 10%左右。采样点剪枝能够大大减少无效的粗采样点数量，虽然略微损失了一些精度，但带来了约 25%的效率提升。此外，完整的改进模型融合了它们的各自优势并实现了最佳性能。

4.6.3 泛化性研究

为了验证所提出的注意力模块在新视角合成任务中的泛化能力，在 Realistic Synthetic 360° 数据集上进行了实验。实验使用了 NeRF 的一种加速版本 EfficientNeRF[62] 作为基线模型，并且单独将注意力模块应用在其中。在表 4.3 中展示的定量结果，直接报告了 EfficientNeRF 的成绩，并且还与相关领域的几个最先进方法进行了比较，包括 NeRF、NSVF[63]、PlenOctree 和 JaxNeRF[64]。

表 4.3 Realistic Synthetic 360°数据集的逐场景定量比较

Table 4.3 Per-scene quantitative comparisons on the Realistic Synthetic 360° dataset

PSNR↑

模型	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship	Mean
NeRF	33.00	25.01	30.13	36.18	32.54	29.62	32.91	28.65	31.01
JaxNeRF	34.08	25.03	30.43	36.92	33.28	29.91	34.53	29.36	31.69
NSVF	33.19	25.18	31.23	**37.14**	32.29	**32.68**	34.27	27.93	31.75
PlenOctree	34.66	**25.31**	30.79	36.79	32.95	29.76	33.97	**29.42**	31.71
EfficientNeRF									31.69
Ours	**34.80**	25.13	**32.03**	36.74	**33.96**	30.30	33.55	28.84	**31.92**

SSIM↑

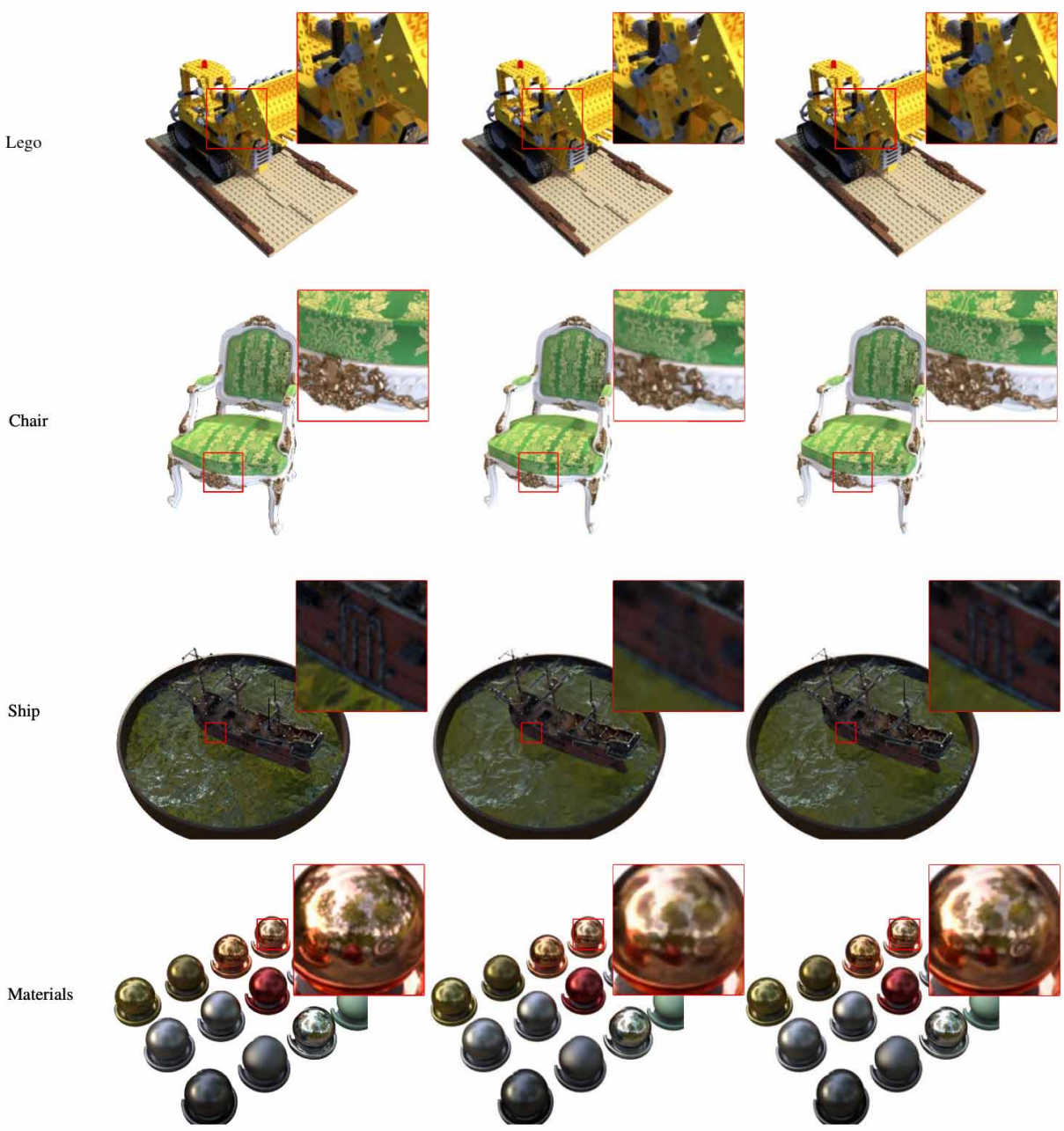
模型	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship	Mean
NeRF	0.967	0.925	0.964	0.974	0.961	0.949	0.980	0.856	0.947
JaxNeRF	0.975	0.925	0.967	0.979	0.968	0.952	**0.987**	0.868	0.953
NSVF	0.968	0.931	0.960	**0.987**	**0.973**	0.853	0.980	**0.973**	0.953
PlenOctree	**0.981**	**0.933**	**0.970**	0.982	0.971	**0.955**	**0.987**	0.884	**0.958**
EfficientNeRF									0.954
Ours	0.974	0.925	0.962	0.978	0.962	0.945	0.980	0.901	0.953

LPIPS↓

模型	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship	Mean
NeRF	0.046	0.091	0.044	0.121	0.050	0.063	0.028	0.206	0.081
JaxNeRF	0.035	0.085	0.038	0.079	0.040	0.060	0.019	0.185	0.068
NSVF	0.043	**0.069**	**0.017**	**0.025**	**0.029**	**0.021**	**0.010**	0.162	0.047
PlenOctree	**0.022**	0.076	0.038	0.032	0.034	0.059	0.017	**0.144**	0.053
EfficientNeRF									**0.028**
Ours	0.024	0.074	0.028	0.030	0.031	0.059	0.015	0.146	0.051

通过定量结果可以看出，在没有其他改进模块的情况下，注意力模块表现出了非常优秀的性能，证明了其泛化能力的卓越性。特别是在 Ficus、Chair 和 Lego 场景中，PSNR 指标分别比对应场景下的最好成绩提高了 0.14、0.8 和 0.68。八个不同场景的平均 PSNR 指标也比基线模型提高了 0.23。此外，改进模型的 SSIM 和 LPIPS 指标也保持了一定的竞争力。需要注意的是，改进模型的超参数设置遵循了 EfficientNeRF 的建议，这可能会限制其发挥最佳性能。

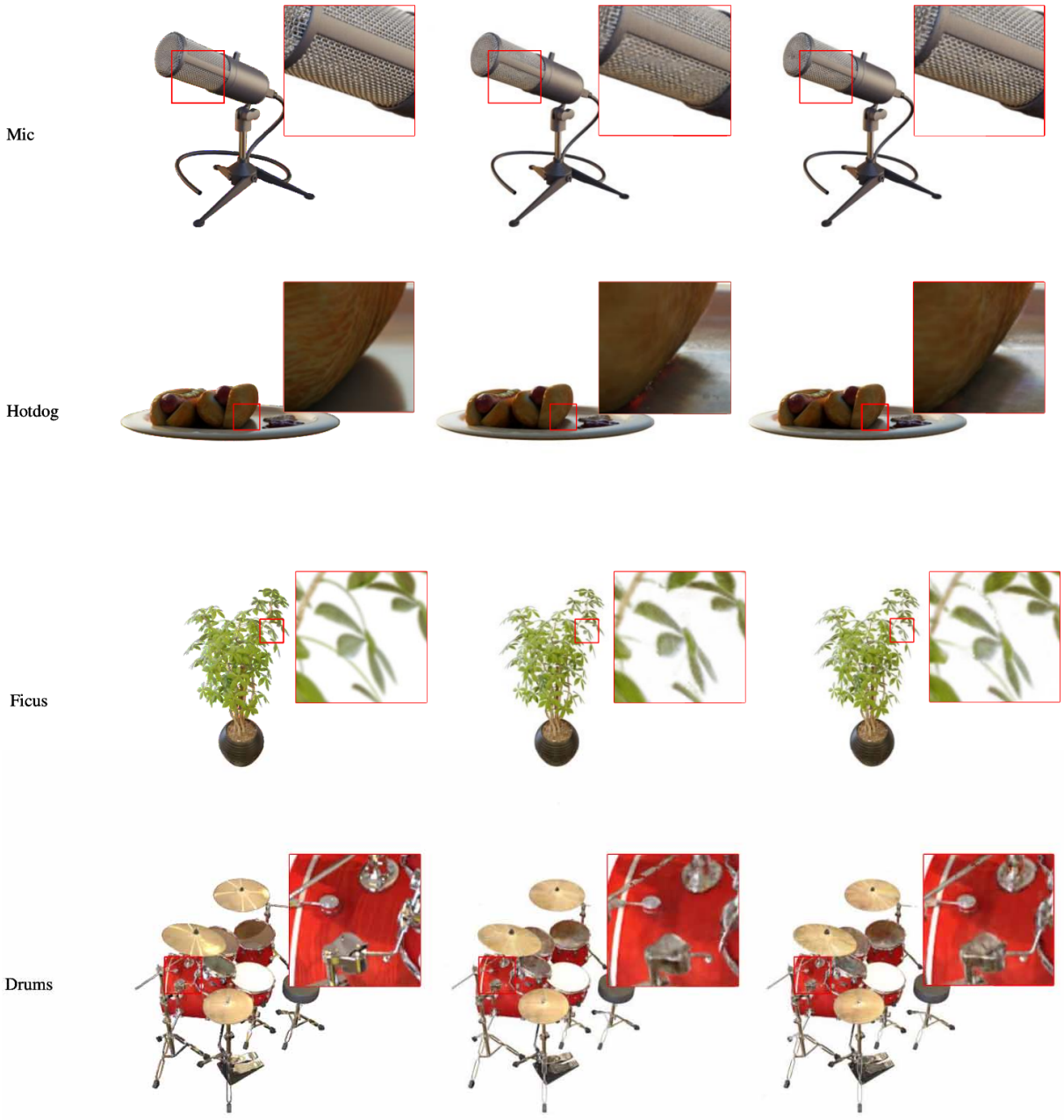
为了进一步验证注意力模块 NeRF 中的有效性，将改进模型的重建结果与基线模型进行了更为直观的定性比较，并展示了每个测试场景下部分视角的渲染图像，如图 4.8 和图 4.9 所示。为了更好地说明渲染结果之间的差异，对每组对比图像的一个特定区域进行了放大。



（a）Ground Truth　　（b）EfficientNeRF　　（c）Ours

图 4.8 Realistic Synthetic 360°数据集中部分场景的定性比较

Figure 4.8 Qualitative comparison of some scenes on the Realistic Synthetic 360° dataset



（a）Ground Truth　　（b）EfficientNeRF　　（c）Ours

图 4.9 Realistic Synthetic 360°数据集中部分场景的定性比较

Figure 4.9 Qualitative comparison of some scenes on the Realistic Synthetic 360° dataset

归功于注意力模块的创新设计，改进模型能够从多个频段的编码信息中准确地捕捉原本被忽视的微小结构和纹理，并减少图像中的高频噪声。在 Lego 场景中，改进模型更好地呈现了黄色积木上部环形凸起的轮廓，使其更加突出清晰，易于区分。在 Chair 场景中，绿色椅面上的金色图案更加复杂和精致。在 Ship 场景中，灰色管道的纹理被成功恢复，使其完整清晰。在 Materials 场景中，各种周围环境在光滑的球形表面上的反

射更加真实，呈现出更自然的外观。在 Hotdog 场景中，热狗盘界面的色彩失真得到了彻底修复。在 Mic 场景中，麦克风的网状保护罩的分辨率得到了提高，外观更加精细。在 Drums 场景中，鼓之间用来固定的金属连杆的完整性得到了提高，不再出现断裂现象。在 Ficus 场景中，细长的绿色叶柄被成功捕捉，使植物的外观更加逼真。

4.7 本章小节

本章的研究探讨了使用高频函数对输入进行编码对网络产生的影响，设计出了一种全新的注意力模块。该模块由全连接层和基于注意力的门控单元组成，能够在捕获 3D 场景中关键的高频信息的同时抑制高频噪声的干扰，有效地捕捉场景中更精细的局部细节。为了实现改进模型的性能与效率的完美平衡，应用了球谐函数来显式策略不同视角下的颜色值，设计了采样点剪枝策略来提高训练的效率以降低训练的时间消耗。在与基线模型 Neus 使用相同的超参数、训练/测试数据集和迭代次数的情况下，改进模型以极少训练时间的增加，换取了更为出色的重建质量。在带掩码和不带掩码的条件下，改进模型的 CD 指标分别降低了 0.1 和 0.06，相比基线模型提升非常显著。此外，所提出的注意力模块没有框架限制，很容易适应其他基于 NeRF 的框架。因此，还在一种最先进的 NeRF 模型中应用了该注意力模块，验证了其泛化能力，并且取得了有竞争力的结果。

5 总结与展望

5.1 总结

针对当前主流的隐式神经表面重建方法仍然存在的问题与挑战，开展了以现有图像和点云这两种不同的输入形式的隐式神经表面重建为基础的改进研究，提出了相应的改进方法，达到了预期的研究目标。

基于点云的隐式神经表面重建的改进研究以 Deepsdf 框架为基础，其主要工作如下：

（1）设计了一个编码器模块，通过对采样点进行高频映射编码和对采样点的局部邻域表面特征进行编码，来提高模型对复杂 3D 对象的精确参数化能力。使用编码模块能够更好地捕捉数据的局部信息，从而提高了重建结果的精度。

（2）设计了自适应的损失加权策略，通过判断不同表面区域的学习难度，并动态调节其权重，来提高模型对训练数据的利用率。该策略在处理非均匀分布的数据时具有明显的优势。

（3）在开源数据集上的实验结果表明，较现有方法显著的提高了重建结果的精度，其重叠度和 F-score 分别提高了 1.458%和 1.46%，平均倒角距离降低了 0.08。

基于图像的隐式神经表面重建的改进研究以 Neus 框架为基础，其主要工作如下：

（1）设计了一种基于 MLP 的注意力模块，引入了新的归纳偏执。该模块提高了重建结果的细节表现能力和表面平滑程度。

（2）引入球谐函数对非朗伯场景下的光场进行显式建模。该方法不仅提高了模型训练效率，而且显著提高了在部分光照复杂的场景下的重建质量。

（3）设计了一种空间粗体素缓存结构来记录场景的 SDF 分布情况。通过查询和更新体素中存储的 SDF 值，能够实现对场景中空白区域的裁剪。该方法在几乎不牺牲重建质量的前提下，显著降低训练的时间消耗。

（4）在开源数据集上的实验表明，较现有方法显著的提高了重建结果的精度和表面的平滑度和还原度。在使用和不使掩码的情况下，其平均倒角距离分别降低了 0.1 和 0.06。

上述以图像和点云这两种不同的输入形式隐式神经表面重建改进均能显著提高相应评价指标的成绩，并提供了更好的视觉效果。

5.2 展望

虽然本文所提出的改进方法已经取得了一定成果，但在特定条件下仍可能存在局限性。因此，需要进一步研究和改进，以使其更加实用和有效。

在基于点云的隐式神经表面重建的改进研究中，使用了 kD-tree 来进行邻居点查找，

虽然在大多数情况下能够提供良好的性能，但是随着数据集的增大，计算资源的负担也越来越大。因此，需要探索更为合理的替代方案，以提高模型的训练效率。在未来的研究中，可以考虑使用基于哈希表的方法来进一步加速该查找过程。

在基于图像的隐式表面重建的改进研究中，提出了一种使用注意力机制对位置编码进行处理的方法。虽然实验结果显示了该方法的有效性，但缺乏对注意力模块起作用的方法进行验证。在未来的工作中，需要探索通过可视化的方法来分析注意力模块对不同频率基上的信号产生的增强和抑制作用。这将有助于更直观地理解注意力模块的作用机制，并为优化其性能提供更多的指导。除此之外，还将进一步研究注意力模块在其他应用场景中的适用性，以拓展其应用范围。

隐式神经表面重建方法已经在许多领域得到广泛应用，如计算机视觉、虚拟现实、机器人学和仿真等。未来，还将继续发挥其优势，扩展到更多的应用场景。同时，关于隐式神经表面重建的研究还有许多挑战需要克服，例如：如何处理不完整、噪声和变形数据，如何提高模型的可解释性和可控性，以及如何更好地结合多源信息进行表面重建等等。这些问题的解决将会推动隐式神经表面重建领域的发展，带来更多新的应用和突破。

参考文献

[1] Mescheder L, Oechsle M, Niemeyer M, et al. Occupancy networks: Learning 3d reconstruction in function space[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 4460-4470.

[2] Park J J, Florence P, Straub J, et al. Deepsdf: Learning continuous signed distance functions for shape representation[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 165-174.

[3] Gropp A, Yariv L, Haim N, et al. Implicit geometric regularization for learning shapes[J]. arXiv preprint arXiv:2002.10099, 2020.

[4] Lacroute P, Levoy M. Fast volume rendering using a shear-warp factorization of the viewing transformation[C]//Proceedings of the 21st annual conference on Computer graphics and interactive techniques. 1994: 451-458.

[5] Sitzmann V, Chan E, Tucker R, et al. Metasdf: Meta-learning signed distance functions[J]. Advances in Neural Information Processing Systems, 2020, 33: 10136-10147.

[6] Duan Y, Zhu H, Wang H, et al. Curriculum deepsdf[C]//Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16. Springer International Publishing, 2020: 51-67.

[7] Martel J N P, Lindell D B, Lin C Z, et al. Acorn: Adaptive coordinate networks for neural scene representation[J]. arXiv preprint arXiv:2105.02788, 2021.

[8] Tretschk E, Tewari A, Golyanik V, et al. Patchnets: Patch-based generalizable deep implicit 3d shape representations[C]//European Conference on Computer Vision. Springer, Cham, 2020: 293-309.

[9] 席建锐,唐红梅,梁春阳,刘鑫.基于改进隐函数的点云物体重建[J/OL].计算机工程:1-11[2022-11-16].DOI:10.19678/j.issn.1000-3428.0064984.

[10] Unser M, Aldroubi A, Eden M. B-spline signal processing. I. Theory[J]. IEEE transactions on signal processing, 1993, 41(2): 821-833.

[11] Graves A, Graves A. Long short-term memory[J]. Supervised sequence labelling with recurrent neural networks, 2012: 37-45.

[12] Mu J, Qiu W, Kortylewski A, et al. A-sdf: Learning disentangled signed distance functions for articulated shape representation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 13001-13011.

[13] Saito S, Huang Z, Natsume R, et al. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 2304-2314.

[14] Mildenhall B, Srinivasan P P, Tancik M, et al. Nerf: Representing scenes as neural radiance fields for view synthesis[J]. Communications of the ACM, 2021, 65(1): 99-106.

[15] 常远,盖孟.基于神经辐射场的视点合成算法综述[J].图学学报,2021,42(03):376-384.
[16] 朱方.3D 场景表征—神经辐射场（NeRF）近期成果综述[J].中国传媒大学学报(自然科学版),2022,29(05):64-77.DOI:10.16196/j.cnki.issn.1673-4793.2022.05.007.
[17] Niemeyer M, Mescheder L, Oechsle M, et al. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 3504-3515.
[18] Yariv L, Gu J, Kasten Y, et al. Volume rendering of neural implicit surfaces[J]. Advances in Neural Information Processing Systems, 2021, 34: 4805-4815.
[19] Jiang Y, Ji D, Han Z, et al. Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 1251-1261.
[20] Oechsle M, Peng S, Geiger A. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 5589-5599.
[21] Wang P, Liu L, Liu Y, et al. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction[J]. arXiv preprint arXiv:2106.10689, 2021.
[22] Delaunay B. Sur la sphere vide[J]. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk, 1934, 7(793-800): 1-2.
[23] 高菲,任鸿翔,闫霞.基于 Delaunay 三角网的大规模航道地形构建及疏密控制[J].水运工程,2022(12):215-220.DOI:10.16233/j.cnki.issn1002-4972.20221129.008.
[24] 李想,张琦,程酉维,殷亚军,沈旭,计效园,周建新.复杂铸件 Delaunay 三角剖分方法及其在华铸 CAE 的应用[J].特种铸造及有色合金,2022,42(11):1351-1354.DOI:10.15980/j.tzzz.2022.11.007.
[25] 侯竞夫.激光雷达数据的 Delaunay 三角网格简化算法研究进展分析[J].电子测试,2022,36(17):44-47+55.DOI:10.16520/j.cnki.1000-8519.2022.17.033.
[26] Burrough P A, McDonnell R A, Lloyd C D. Principles of geographical information systems[M]. Oxford university press, 2015.
[27] 王径舟,欧阳润海.基于 Voronoi 图结构描述的有机无机杂化钙钛矿带隙机器学习[J].硅酸盐学报,2023,51(02):397-404.DOI:10.14062/j.issn.0454-5648.20220945.
[28] 和仕芳,张方浩,杜浩国,曹彦波.加权泰森多边形在 2021 年云南漾濞 M_S6.4 地震后应急避险安置点责任区划分的应用[J].地震研究,2023,46(01):128-137.DOI:10.20015/j.cnki.issn1000-0666.2023.0014.
[29] Hoppe H, DeRose T, Duchamp T, et al. Surface reconstruction from unorganized points[C]//Proceedings of the 19th annual conference on computer graphics and interactive techniques. 1992: 71-78.
[30] Curless B, Levoy M. A volumetric method for building complex models from range

images[C]//Proceedings of the 23rd annual conference on Computer graphics and interactive techniques. 1996: 303-312.

[31] Carr J C, Beatson R K, Cherrie J B, et al. Reconstruction and representation of 3D objects with radial basis functions[C]//Proceedings of the 28th annual conference on Computer graphics and interactive techniques. 2001: 67-76.

[32] Kolluri R. Provably good moving least squares[J]. ACM Transactions on Algorithms (TALG), 2008, 4(2): 1-25.

[33] Kazhdan M, Bolitho M, Hoppe H. Poisson surface reconstruction[C]//Proceedings of the fourth Eurographics symposium on Geometry processing. 2006, 7: 0.

[34] 杨友宁. 点云的泊松曲面重建方法研究与实现[D]. 中北大学,2022.DOI:10.27470/d.cnki.ghbgc.2022.000092.

[35] 鲁猛胜. 基于法向约束和屏蔽泊松方程的点云表面重建[D]. 武汉大学,2020.DOI:10.27379/d.cnki.gwhdu.2020.000908.

[36] Lorensen W E, Cline H E. Marching cubes: A high resolution 3D surface construction algorithm[J]. ACM siggraph computer graphics, 1987, 21(4): 163-169.

[37] Max N. Optical models for direct volume rendering[J]. IEEE Transactions on Visualization and Computer Graphics, 1995, 1(2): 99-108.

[38] Max N, Chen M. Local and global illumination in the volume rendering integral[R]. Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2005.

[39] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators[J]. Neural networks, 1989, 2(5): 359-366.

[40] Qi C R, Su H, Mo K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 652-660.

[41] Rahaman N, Baratin A, Arpit D, et al. On the spectral bias of neural networks[C]//International Conference on Machine Learning. PMLR, 2019: 5301-5310.

[42] Mildenhall B, Srinivasan P P, Tancik M, et al. Nerf: Representing scenes as neural radiance fields for view synthesis[J]. Communications of the ACM, 2021, 65(1): 99-106.

[43] Chang A X, Funkhouser T, Guibas L, et al. Shapenet: An information-rich 3d model repository[J]. arXiv preprint arXiv:1512.03012, 2015.

[44] Stutz D, Geiger A. Learning 3d shape completion under weak supervision[J]. International Journal of Computer Vision, 2020, 128(5): 1162-1181.

[45] Downs L, Francis A, Koenig N, et al. Google scanned objects: A high-quality dataset of 3d scanned household items[C]//2022 International Conference on Robotics and Automation (ICRA). IEEE, 2022: 2553-2560.

[46] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

[47] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[48] Melas-Kyriazi L. Do you even need attention? a stack of feed-forward layers does surprisingly well on imagenet[J]. arXiv preprint arXiv:2105.02723, 2021.

[49] Tolstikhin I O, Houlsby N, Kolesnikov A, et al. Mlp-mixer: An all-mlp architecture for vision[J]. Advances in neural information processing systems, 2021, 34: 24261-24272.

[50] Ding X, Xia C, Zhang X, et al. Repmlp: Re-parameterizing convolutions into fully-connected layers for image recognition[J]. arXiv preprint arXiv:2105.01883, 2021.

[51] Liu H, Dai Z, So D, et al. Pay attention to mlps[J]. Advances in Neural Information Processing Systems, 2021, 34: 9204-9215.

[52] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[53] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.

[54] Hinton G E. Training products of experts by minimizing contrastive divergence[J]. Neural computation, 2002, 14(8): 1771-1800.

[55] Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[C]//Proceedings of the 27th international conference on machine learning (ICML-10). 2010: 807-814.

[56] Hendrycks D, Gimpel K. Gaussian error linear units (gelus)[J]. arXiv preprint arXiv:1606.08415, 2016.

[57] Green R. Spherical harmonic lighting: The gritty details[C]//Archives of the game developers conference. 2003, 56: 4.

[58] Yu A, Li R, Tancik M, et al. Plenoctrees for real-time rendering of neural radiance fields[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 5752-5761.

[59] Jensen R, Dahl A, Vogiatzis G, et al. Large scale multi-view stereopsis evaluation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 406-413.

[60] Yao Y, Luo Z, Li S, et al. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 1790-1799.

[61] Schonberger J L, Frahm J M. Structure-from-motion revisited[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 4104-4113.

[62] Hu T, Liu S, Chen Y, et al. Efficientnerf efficient neural radiance fields[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 12902-12911.

[63] Liu L, Gu J, Zaw Lin K, et al. Neural sparse voxel fields[J]. Advances in Neural Information Processing Systems, 2020, 33: 15651-15663.

[64] Deng B, Barron J T, Srinivasan P P. JaxNeRF: an efficient JAX implementation of NeRF[J]. 2020.