用于三维网格压缩的分层神经曲面

Sai Karthikey Pentapati*、Gregoire Phillips†、Alan C. Bovik**图像
与视频工程实验室，德克萨斯大学奥斯汀分校
†爱立信研究院，加利福尼亚州圣克拉拉

arXiv:2512.15985v1 [cs.CG] 17 Dec 2025

摘要—隐式神经表示已被证明能够对图像、视频、三维曲面和音频等多种模态实现最先进的压缩。大多数研究都聚焦于构建传统三维几何体隐式表示（如有符号距离函数）的神经对应形式。然而，基于三角网格的几何表示仍是工业界最广泛使用的表示形式，而能够生成此类表示的隐式神经表示鲜有研究。本文提出一种用于构建零亏格三维流形紧凑隐式神经表示的方法。该方法首先为给定的三维网格创建球面参数化——将网格曲面映射到单位球面的曲面上——然后构建一种隐式神经表示，对其曲面上连续定义的位移矢量场进行编码，以重建原始形状。该表示的紧凑性得益于其层次结构：它先恢复编码曲面的粗略结构，再为其添加高频细节。计算出隐式神经表示后，即可从中解码出任意分辨率和连通关系的三维网格。解码可实时完成，同时在重建质量与压缩表示大小之间实现最先进的权衡。

索引词—三维网格压缩、隐式神经表示、神经几何处理、球谐函数

I. 引言

近年来，尽管三维场景领域涌现出许多新颖的表示方式，三维网格仍是应用最广泛的表示形式，因为它能够紧凑且连续地表示几何，便于存储额外的表面属性（如纹理贴图、法线贴图、凹凸贴图等），并且易于利用当前的图形处理硬件进行渲染。随着虚拟现实、增强现实和混合现实应用场景的兴起，三维媒体的紧凑存储与高效分发变得前所未有地重要。传统上，研究主要聚焦于压缩网格：要么无损压缩连通关系 [19], [20] 并量化几何（如 Google Draco[21] 等工业标准所采用的方法），要么通过减少顶点数 [22],[23]来简化网格。近年来，神经压缩方法日益受到关注，其中隐式神经表示（INR）[1] 尤为引人瞩目。

隐式神经表示具有多种特性，使其在压缩应用中的研究颇具吸引力。首先，与基于像素或体素的表示不同，隐式神经表示的大小与分辨率无关，仅取决于底层信号的复杂程度，因此可以在无限分辨率下进行查询。尽管编码后函数的带宽受所用神经网络容量的限制，但隐式神经表示在任意点进行查询时所固有的插值往往具有更强的非线性，并且比使用双三次插值、Lanczos、重心插值等方法对传统离散表示进行插值时依赖更大的感受野；在后者中，感受野由核大小固定，因此隐式神经表示能够实现更有意义的采样。隐式神经表示还具有无限阶可微性，这与将上述分段插值方案应用于离散数据后得到的连续信号不同；当需要直接应用连续信号处理算法时，这一特性可能很有用，尤其是在离散三维几何处理中，因为众所周知的“免费午餐”定理表明并不存在适用于所有情况的最佳方法。此外，由于采用逐实例训练，隐式神经表示比传统压缩技术更能保留高频细节。凭借这些优势，许多研究构建了隐式神经表示，用于填补缺失部分、插值和压缩不同类型的信号，例如图像、视频、三维形状的有符号距离函数，以及辐射场。

近年来，许多研究致力于构建三维形状的紧凑神经表示。ACORN [11] 和 NGLOD[12] 分别采用基于八叉树的空间划分，以实现占据场和有符号距离函数的内存高效表示。隐式位移场（IDF） [8] 将 SDF 编码为两个隐式神经表示的组合：第一个编码粗略形状的 SDF，第二个则建模一个位移场，为粗略曲面添加细节。尽管这些方法能够实现出色的重建质量，但它们并非内存最优，因为神经网络必须将宝贵的学习能力耗费在不位于曲面上的点上。这是因为 SDF/占据场需要在三维曲面周围环境空间中的任意点上都有良好定义（基于八叉树的/层次表示在一定程度上缓解了这一问题，但并不彻底）。此外，这些网络还必须经过正则化，以遵守 SDF 的程函方程，确保其梯度的幅值为单位值。从这些隐式神经表示中重建编码曲面通常也较为缓慢，因为这需要使用类似行进立方体 [13] 的算法，从编码后的 SDF 中提取等值面。使用行进立方体提取的网格往往存在拓扑不一致性，例如不具备水密性，并且还包含视觉上令人不适的离散化伪影。

为规避这些问题，一些方法提出将非常粗糙的表面模型显式地存储为压缩表示的一部分

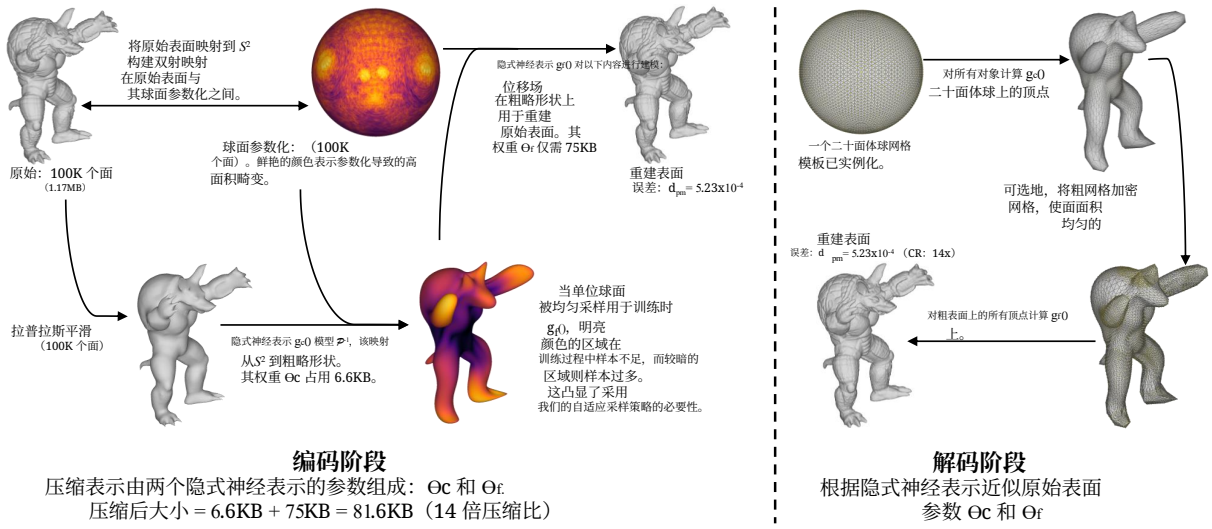

图 1。**我们用于网格压缩的方法概述：**给定曲面表示为其球面参数化的逆函数，随后由层次化隐式神经表示进行建模。该方法即使在高压缩比下也能实现高质量重建。

，并使用以隐式神经表示编码的位移场为其添加细节，从而避免对基于 SDF 的表示进行昂贵的等值面提取。其他方法通过先将曲面参数化到预定义的图表（例如单位圆 [16]或单位球面 [17], ），再训练一个表示该参数化逆函数的隐式神经表示来解决这一问题。与基于 SDF 的方法不同，这些方法能够实时直接生成网格，且不会产生拓扑伪影。前一类方法的显式-隐式混合表示无法推广到存储网格的表面属性，例如纹理贴图、法线贴图等；而后一类方法则会产生变形伪影，因为将曲面参数化为更简单的形状时，往往无法保留三角形的结构（面积比、内角等）。

本文提出了一种压缩三维网格的方法，该方法属于后一种范式，但融入了层次化流程。层次化策略使我们能够在隐式神经表示训练过程中实现一种新颖的自适应采样技术，从而加快训练速度，并通过在很大程度上减轻变形伪影来提升重建质量，使我们的方法能够实现最先进的三维网格压缩。我们的表示还具有无限可微性，并且可以扩展到压缩表面贴图；相比之下，目前领先的网格压缩方法采用的是混合显式-隐式结构。

II. 方法

A. 概述

图 1 展示了我们方法的整体流程。形式化地，考虑一个三维网格曲面 $\mathcal{M}(V,\ F) \subset \mathbb{R}^3$，其包含顶点 $V \in \mathbb{R}^3$ 和面 F。其编码过程可以分解为以下步骤：

1) 通过将所有顶点 $\mathcal{P}(V) = V_s$ 单射到单位球面上，生成 $\mathcal{P}$ 的球面参数化 $\mathcal{M}$

同时保持相同的连通关系 F及其拓扑，以获得 $\mathcal{M}_s(V_s, F)$。我们采用 Praun 和 Hoppe [24]提出的算法

以执行此映射。2）首先将 $\mathcal{M}$上的任意曲面点表示为包含该点的三角形顶点的重心插值，然后使用相同的重心坐标对 $\mathcal{M}_s$中同一面上的顶点进行插值，从而将此单射扩展为双射 $\mathcal{P}:\ \mathcal{M} \leftrightarrow \mathcal{M}_s$。3）对 $\mathcal{M}$ 应用 c 次拉普拉斯平滑 [25]，生成其粗化版本 $\mathcal{M}_c$。需要注意的是，$\mathcal{M}$，$\mathcal{M}_s$和 $\mathcal{M}_c$ 具有相同的连通关系和拓扑（F）。因此，无需显式重新计算 $\mathcal{M}_c$的球面参数化，也可以定义双射 $\mathcal{P}_c:\ \mathcal{M}_c \leftrightarrow \mathcal{M}_s$。4）通过以类似于 [17]的方式，使用参数为 θ_c 的小型多层感知器[26] q_c（·）进行训练，来近似参数化 $\mathcal{P}_c^{-1}$：$\mathcal{M}_s \to \mathcal{M}_c$的逆映射。由此构建出 $\mathcal{P}_c^{-1}$的隐式神经表示（INR），它可以根据 S^2生成 $\mathcal{M}_c$的近似结果 $\mathcal{M}_c^*$。更多详情请参见第 II-B 节。5）使用第二个多层感知器（MLP） q_f（·），将更精细的高分辨率细节添加到 $\mathcal{M}_c^*$ 中。该感知器的参数为 θ_f，构成一个隐式神经表示，用于编码位移场 $\mathcal{D}:\ \mathcal{M}_c^* \to \mathbb{R}^3$，使得 $\mathcal{M}^* := \mathcal{M}_c^* + \mathcal{D}(\mathcal{M}_c^*)$ 能够很好地近似 $\mathcal{M}$。第 II-C 节详细介绍了如何使用我们的自适应采样对其进行训练。

两个隐式神经表示 θ_c 和 θ_f 的参数构成我们的压缩表示。可选地，还可以对这些参数进行剪枝、量化和熵编码，以实现进一步压缩。连通关系信息 F 不包含在压缩表示中。解码器按照以下方式从该层次化隐式神经表示中重建网格：

1) 初始化一个 $\mathcal{I}_k$（V_k，F_k）级别为 k 的二十面体球网格。

从一个单位二十面体开始，通过迭代细分并将新引入的顶点投影到单位球面上 k 次，即可创建级别为 k 的二十面体球网格。

2) 通过计算$V_c^* = q_c(V_k)$，将所有顶点 V_k 投影到粗表面上。由此得到 $\mathcal{M}_c^*(V_c^*,\ F_k)$。

3) 可选地，判断 $\mathcal{M}_c^*$ 中的某些面是否显著大于其余面。我们通过启发式方法找出所有面积超过面面积中位数四倍的面。随后细分 $\mathcal{I}_k$ 中对应的面，并将新引入的顶点通过 q_c 进行变换，从而获得面面积更加均匀的 $\mathcal{M}_c^*$。这一过程可以重复进行，以得到重建拓扑 F_k^*，直至启发式方法不再识别出任何面。

4) 计算重建粗网格所有顶点处的位移场，并将其应用于相同的顶点：$V^* = q_f(V_s^*) + V_s^*$。所得重建结果$\mathcal{M}^*(V^*,\ F_k^*)$近似于原始网格 $\mathcal{M}(V,\ F)$。

原始网格 $\mathcal{M}$ 可能附带属性映射 $\mathcal{S}(p)$：$\mathcal{M} \to \mathbb{R}^d$, $p \in \mathcal{M}$ 例如纹理、法线或凹凸贴图。其编码过程可以采用类似的方法完成，但为简洁起见，未在上述表述中展开。

B. 粗层隐式神经表示训练

训练用于重建粗表面的隐式神经表示的过程大体遵循 [17]。首先，在 $\mathcal{M}_s$上均匀采样点。具体而言，可以先为 $\mathcal{M}_s$ 中的每个面分配一个与其面积成正比的概率，然后根据所得分布抽取面，最后在这些面上随机选择点。随后，将采样点归一化为单位长度，以确保它们位于 S^2上。（之所以需要这一步，是因为尽管采样点已经非常接近 S^2，因为 $\mathcal{M}_s$是 $\mathcal{M}$的球面参数化，但由于 $\mathcal{M}_s$具有离散网格的性质，这些点可能并不完全位于其上。）使用第二步编码过程中所述的相同重心插值技巧，对所有采样点计算 $\mathcal{P}_c^{-1}$。通过最小化期望 $\mathrm{E}_{p\in S^2}(||q_c(p) - \mathcal{P}_c^{-1}(p)||_2^2)$，这些计算得到的样本可作为训练隐式神经表示 q_c 的真实值，其中 p从 S^2中均匀采样。

C. 细层隐式神经表示训练

采用与粗层训练类似的批量采样策略存在局限。均匀采样 $\mathcal{M}_s$ 并计算其对应的 q_c，无法在 $\mathcal{M}_c^*$上得到均匀分布的样本。因此，均匀采样 $\mathcal{M}_s$ 用于训练 q_f，会导致隐式神经表示的模型容量分配效率低下。为缓解这一问题，我们引入了一种由表示的层次结构支持的自适应采样策略。

前置知识——度量张量： 由于 q_c 是从$S^2 \to \mathbb{R}^3$到 S^2 的映射，而 S^2 还可以通过函数复合，仅用两个变量——方位角 u和极角 v——进一步参数化，$q(u,\ v) := q_c(S^2(u,\ v))$，因此可以得到一个

表 I 从 THINGI10K 数据集中选取的 15 个网格压缩至不同目标尺寸时的平均值 $d_{pm} \times 10^4$ 和 d_n（°）

方法	50KB	85KB	165KB	260KB
Ours	6.56 / 7.12	4.80 / 5.69	3.11 / 3.20	2.41 / 3.23
QNDF	9.12 / 10.67	6.92 / 6.52	3.93 / 3.90	2.43 / 3.34
SNS（量化）	16.31 / 13.89	11.77 / 9.15	7.07 / 7.41	5.14 / 5.18
我们的方法（NQ）	13.11 / 9.91	9.05 / 7.08	5.98 / 4.87	5.42 / 4.69
QNDF(NQ)	15.75 / 13.26	11.98 / 7.99	6.88 / 5.41	5.30 / 4.69
SNS	20.12 / 19.31	14.42 / 12.54	10.40 / 9.43	8.01 / 7.87
QS-DRC	24.41 / 17.92	15.05 / 11.99	7.23 / 4.70	3.18 / 3.44

从 $\mathbb{R}^2$ 到 $\mathbb{R}^3$ 的映射。该映射的度量张量 $\mathrm{I}(u,\ v)$（也称为第一基本形式）在参数空间的每个点处定义如下：

$$\mathrm{I}(u,v) = \begin{bmatrix} \frac{\partial q}{\partial u}\cdot\frac{\partial q}{\partial u} & \frac{\partial q}{\partial u}\cdot\frac{\partial q}{\partial v} \\ \frac{\partial q}{\partial u}\cdot\frac{\partial q}{\partial v} & \frac{\partial q}{\partial v}\cdot\frac{\partial q}{\partial v} \end{bmatrix} \tag{1}$$

随后，畸变比 $d(u,\ v) = \sqrt{\det(\mathrm{I}(u,\ v))}/\sin^2(v)$ 给出了映射 q 作用于 S^2 上的 $(u,\ v)$ 点时，无穷小面积的缩放因子。由于使用神经网络 q_c 来表示这一映射，它处处连续可微，并且可以在任意点计算其度量张量。

最后，训练 q_f 所需的批次可以按以下方式生成：

1）实例化一个细分层级相对较低的二十面体球。2）在其每个面中心处计算畸变比 d。这两个步骤只需在训练开始前执行一次。3）根据各面的畸变比按比例设定面概率，从二十面体球中抽取样本 p 。4）使用相同的重心插值技巧，计算这些点对应的 $\mathcal{P}^{-1}(p)$，从而获得位于原始网格 $\mathcal{M}$ 上的对应点。5）计算这些点处的位移场 $\mathcal{D}(p) = \mathcal{P}^{-1}(p) - q_c(p)$，并将其用作训练目标。

最后，通过最小化 $\mathrm{E}_{p\in S^2}(||q_f(q_c(p)) - \mathcal{D}(p)||_2^2)$ 来训练 q_f，其中 p 按照前述策略从 S^2 中采样。在此阶段，不会优化 q_c 的参数。这种数据采样策略紧密模拟了解码器执行的网格重建过程，因此提升了压缩效果。

D. 模型架构

我们使用带有残差连接和 SiLU激活函数 [27] 的多层感知机对 q_c 和 q_f 进行建模。q_c 的输入是一个三维坐标（$x,\ y,\ z$）$\in S^2$，输出则是粗网格 $\mathcal{M}_c^*$ 上估计的三维位置。另一方面，我们不直接将 $\mathcal{M}_c^*$ 上的三维坐标传递给 q_f，而是首先以与 [10] 相同的方式对其进行位置编码，以克服神经网络的频谱偏差 [28]。对 q_c 的输入进行位置编码既没有必要，甚至是不理想的，原因在于：(a) 它不需要编码非常高频的信号；(b) 容易发生过拟合；(c) 必须增加 MLP 第一

层中的参数数量，从而增大压缩表示的规模。层数和每层的隐藏单元数是决定压缩比的超参数。我们在第三节详细介绍了模型规模与重建质量之间的变化。为获得更好的压缩结果，我们在训练后将 MLP 的参数量化为 16 位浮点数，方法类似于 [15]。

III. 实验

我们在 Thingi10k 数据集中的 15 个水密零亏格网格上评估了我们的方法 [33]，这些网格涵盖了各种复杂几何体，顶点数也各不相同。我们将每个网格压缩为四种大小（50KB、85KB、165KB 和 260KB），并与以下基线方法进行了比较：

1) 量化神经位移场（QNDF） [15]：一种混合显式—隐式隐式神经表示，在待编码曲面的简化版本上对位移场进行建模，是当前三维网格压缩领域最先进的隐式神经表示方法。2) 球面神经曲面（SNS） [17]：我们这种表示方法的非层次化版本。我们将其方法中使用的激活函数和调度器替换为我们选用的配置，使其性能略有提升。3) Q-SLIM [22] + Google Draco 与量化 [21]：一种具有竞争力的基于非隐式神经表示的有损压缩基线方法，可在交互式时间内完成编码。

所有基于隐式神经表示的方法的结果均展示了模型参数量化前后的情况。我们使用平均点到网格误差和平均法线误差来衡量所有方法在各比特率下的重建质量，结果见表 I。图 2 还展示了定性比较。与现有方法相比，我们的方法在高压缩比下取得了更显著的改进。在较低压缩比下，我们发现，非隐式神经表示方法能够在实时运行的同时实现相当的重建质量。

A. 训练细节

对于所有比特率，我们的多层感知器 q_c 包含 20 个隐藏层，每层有 12 个隐藏节点。将其参数量化为 16 位后，模型大小为 6.6KB；否则为 12.2KB。同样，在将三维坐标输入第二个多层感知器 q_f之前，我们对其进行位置编码，频率层级最高为 10。两个模型 q_c 和 q_f 的训练批量大小均设为 2048。q_c 训练了 50,000 次迭代（耗时 ~ 8 分钟），而 q_f 使用 AdamW 优化器和余弦退火，以 10^{-3} 的学习率训练了 15,000 次迭代。我们的超参数选择

q_f 不同比特率模型的训练时间和图形处理器使用情况、使用 8 级二十面体球模板时的解码时间和图形处理器使用情况，以及作为输入的内容均列于表 II。使用我们的方法，每个形状的总编码时间为 25 至 35 分钟，与其他基于隐式神经表示的压缩方法（如 QNDF 和 [14]）相当。我们的分层方法也使我们的方法比 SNS 更快；SNS 的平均训练时间为 8 至 10 小时。

尽管使用我们的方法编码网格耗时较长，但包含 327K 个顶点的网格可在 100 毫秒内解码。这比基于 SDF 的神经表示快几个数量级（每个形状需 2 至 4 分钟），与基于隐式神经表示的网格压缩方法（如 QNDF）相近。所有实验均在 NVIDIA RTX 2070 图形处理器上进行。

编码后的网格结果和我们方法的代码可以在我们的项目页面上找到。

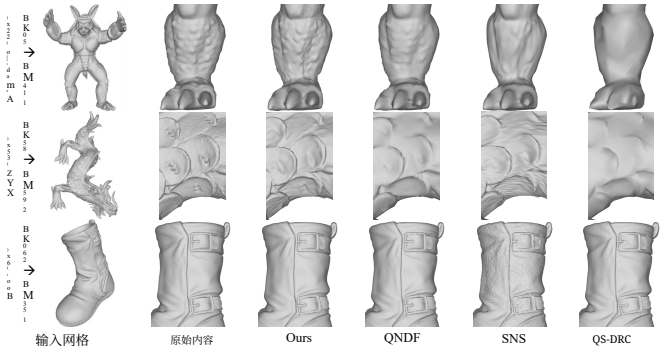

图 2。我们的方法能够在广泛的比特率范围内准确重建原始网格。QNDF 存在明显的量化伪影，SNS 会产生变形伪影，而 QS-DRC 生成的分辨率网格非常低

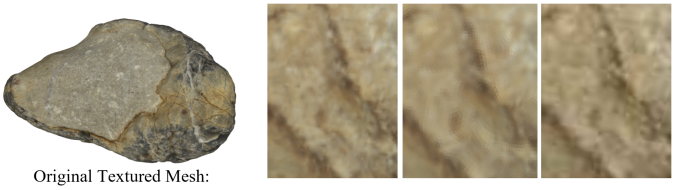

图 3。我们的方法还可以成功扩展到压缩纹理等表面贴图。

表 II 不同目标尺寸的训练参数

压缩后大小	隐藏层数	隐藏单元数	训练时间	显存占用	解码时间
50 KB	18	36	14 分钟	320MB	230ms
85KB	20	44	14 分钟	360MB	280毫秒
165KB	24	58	17分钟	480MB	330毫秒
260KB	28	68	20分钟	600MB	440毫秒

B. 补充实验

图3展示了一个对尺寸为 1024×1024 的纹理图进行编码的示例。将带有重建纹理的网格渲染为图像，并标出了其相对于使用原始纹理渲染的图像的峰值信噪比（PSNR）和结构相似性（SSIM）[32] 。我们的方法性能与 JPEG 压缩相当，且内存占用略低 $qp = 50$ 。请注意，存储 JPEG纹理所需的内存还包括存储相应 UV参数化所需的内存，而我们的表示则无需如此。此外，由于我们的表示本质上与 [17] 相同，只是采用了层次结构，因此我们也可以以类似的方式对其进行微分运算。

IV. 结论

本文提出了一种构建紧凑型隐式神经表示（INR）的方法，用于表示三维网格，并取得了最先进的压缩效果。与以往的三维网格压缩方法不同，我们还展示了对曲面属性进行编码的方法。此外，由于我们的表示具有完全可微的特性，我们还可以直接从该表示中计算编码曲面的微分量。

由于需要逐实例训练，我们的编码速度较慢，限制了该方法在实时应用场景中的使用。未来在这一方向上的研究极具意义，将混合显式-隐式预训练编码器（例如[29]）扩展到球面形式，可能是一个很有前景的方向。此外，我们仅展示了该方法在零属曲面上的应用。通过选择合适的模板形状和参数化算法，理论上可以将其扩展到任意属，但我们将此留待未来工作。

我们认为，我们的工作展示了神经表示在压缩三维几何体方面的有效性与灵活性，并推动进一步研究，以加速和改进这些表示，使其能够切实应用于工业工作流程。

参考文献

[1] Vincent Sitzmann、Julien N. P. Martel、Alexander W. Bergman、David B. Lindell 和 Gordon Wetzstein。2020年。具有周期性激活函数的隐式神经表示。载于《神经信息处理系统》第34届国际会议论文集。[2] Duchon, C. E., 1979年：一维和二维兰索斯滤波。《应用气象与气候学杂志》，18，1016–1022[3] Yannick Str¨umpler、Janis Postels、Ren Yang、Luc Van Gool 和 Federico Tombari。2022年。用于图像压缩的隐式神经表示。载于《计算机视觉——ECCV 2022：第17届欧洲会议》。[4] V. Saragadam 等，“WIRE：小波隐式神经表示”，2023年 IEEE/CVF计算机视觉与模式识别会议（CVPR），加拿大不列颠哥伦比亚省温哥华，2023年，第18507-18516页，doi：10.1109/CVPR52729.2023.01775。[5] Z. Chen 等，“VideoINR：为连续时空超分辨率学习视频隐式神经表示”，2022年 IEEE/CVF计算机视觉与模式识别会议（CVPR），第2037-2047页，doi：10.1109/CVPR52688.2022.00209。[6] H. Chen、M. Gwilliam、S. -N. Lim 和 A. Shrivastava，“HNeRV：视频混合神经表示”，2023年 IEEE/CVF计算机视觉与模式识别会议（CVPR），加拿大不列颠哥伦比亚省温哥华，2023年，第10270-10279页，doi：10.1109/CVPR52729.2023.00990。[7] J. J. Park、P. Florence、J. Straub、R. Newcombe 和 S. Love-grove，“DeepSDF：学习用于形状表示的连续有符号距离函数”，2019年 IEEE/CVF计算机视觉与模式识别会议，2019年，第165-174页，doi：10.1109/CVPR.2019.00025。[8] Yifan、Wang、Rahmann、Lukas 和 Sorkine-Hornung、Olga。（2021）。利用隐式位移场实现几何一致的神经形状表示。10.48550/arXiv.2106.05187。[9] V. Saragadam 等。2022年。MINER：多尺度隐式神经表示。载于《计算机视觉——ECCV 2022：第17届欧洲会议》，以色列特拉维夫，2022年10月23日至27日，论文集，第XXIII部分。 318 333 10 1007 978 3 031 20050 2 19 – 。https://doi.org/ 。/ [10] - - - - Ben Mildenhall 、 Pratul P. Srinivasan 、 Matthew Tancik 、 Jonathan T. Barron、Ravi Ramamoorthi 和 Ren Ng。2021年。NeRF：将场景表示为用于视图合成的神经辐射场。《美国计算机学会通讯》65，1 2022 99 106 10 1145（1月），– 。https://doi.org/ 。/ 3503250 – [11] Julien N. P. Martel、David B. Lindell、Connor Z. Lin、Eric R. Chan、Marco Monteiro 和 Gordon Wetzstein。2021年。ACORN：用于神经场景表示的自适应坐标网络。《ACM图形学汇刊》40，4，58 10 1145 3450626 3459785 文章。https://doi.org/ 。/ 。

[12] T. Takikawa 等，“神经几何细节层次：利用隐式三维形状进行实时渲染”，2021年 IEEE/CVF计算机视觉与模式识别会议（CVPR），美国田纳西州纳什维尔，2021年，第11353-11362页，doi：10.1109/CVPR46437.2021.01120。[13] William E. Lorensen 和 Harvey E. Cline。1987年。行进立方体：一种高分辨率三维表面构造算法。SIGGRAPH计算机图形学。 21, 4 1987 163 169 10 1145 37402 37422 （7月），– 。 [14] https://doi.org/ 。/ 。Venkataram Edavamadathil Sivaram、Tzu-Mao Li 和 Ravi Ramamoorthi。2024年。网格神经几何场。载于ACM SIGGRAPH 2024会议论文集（SIGGRAPH ’24）。文章29，1–11。https://doi.org/ 10 1145 3641519 3657399 。/ 。Sai Karthikey Pentapati、 [15] Gregoire Phillips 和 Alan C. Bovik。2025年。基于量化神经位移场的网格压缩。《计算机图形学论坛》（ 2025 10 1111 70074 ）。DOI: [16] https://doi.org/ 。/cgf。L. Morreale、N. Aigerman、P. Guerrero、V. G. Kim 和 N. J. Mitra，“神经卷积曲面”，2022年 IEEE/CVF计算机视觉与模式识别会议（CVPR），美国路易斯安那州新奥尔良，2022年，第19311-19320页，doi：10.1109/CVPR52688.2022.01873。[17] Romy Williamson 和 Niloy J. Mitra。2025年。通过球面神经曲面进行神经几何处理。《计算机图形学论坛》（2025）。DOI：https://doi.org/ 10 1111 70021 。/cgf。Max [18] Wardetzky、Saurabh Mathur、Felix K¨alberer 和 Eitan Grinspun。2007年。离散拉普拉斯算子：没有免费的午餐。载于第五届Eurographics几何处理研讨会论文集（SGP ’07）。Eurographics协会，德国戈斯拉尔，33–37。[19] J. Rossignac，“Edgebreaker：三角网格的连通性压缩”，载于《IEEE可视化与计算机图形学汇刊》，第5卷，第1期，第47-61页，1999年1月至3月，doi：10.1109/2945.764870。[20] K. Mamou、T. Zaharia 和 F. Prˆeteux，“一种用于低复杂度三维网格压缩的基于三角扇的方法”，2009年第16届IEEE图像处理国际会议（ICIP），埃及开罗，2009年，第3513-3516页，doi：10.1109/ICIP.2009.5414075。[21] Galligan F.、Hemmer M.、Strava O.、Zhang F.、Brettle J.：Google/draco：用于压缩和解压缩三维几何网格与点云的库，2018年。网址：https://google.github.io/draco/ [22] Michael Garland 和 Paul S. Heckbert。1997年。使用二次误差度量进行表面简化。载于第24届计算机图形学与交互技术年度会议论文集（SIGGRAPH ’97）。209–216。https://doi.org/10.1145/258734.258849。[23] G. Zhou、S. Yuan 和 S. Luo，“基于二次误差度量与三角形折叠的网格简化算法”，载于《IEEE Access》，第8卷，第196341-196350页，2020年，doi：10.1109/ACCESS.2020.3034075。[24] Emil Praun 和 Hugues Hoppe。2003年。球面参数化与重新网格化。《ACM图形学汇刊》22，3（2003年7月），340–349。https://doi.org/ 10 1145 882262 882274 。/ 。Mathieu Desbrun、Mark Meyer、 [25] Peter Schr¨oder 和 Alan H. Barr。1999年。利用扩散与曲率流对非规则网格进行隐式平滑。载于第26届计算机图形学与交互技术年度会议论文集（SIGGRAPH ’99），https://doi.org/ 10 1145 311535 311576 。/ 。Rumelhart、D.、 [26] Hinton、G. 和 Williams、R.。通过误差反向传播学习表示。《自然》323，533–536（1986）。https://doi.org/ 10 1038 323533 0 。/ a Dan [27] Hendrycks、Kevin Gimpel。2016年。高斯误差线性单元，https://doi.org/ 10 48550 1606 08415 。/arXiv. 。Rahaman、N.、Baratin、 [28] A.、Arpit、D.、Draxler、F.、Lin、M.、Hamprecht、F.、Bengio、Y. 和 Courville、A.。（2019）。论神经网络的频谱偏差。第36届国际机器学习会议论文集，载于《机器学习研究论文集》97：5301-5310。[29] Jiang、Hanwen 和 Yang、Haitao 和 Pavlakos、Georgios 和 Huang、Qixing（2025）。CoFie：利用坐标场学习紧凑的神经曲面表示。arXiv预印本 arXiv: 2406 03417 。Loshchilov I、 [30] Hutter F。解耦权重衰减正则化。arXiv预印本 arXiv:1711.05101。2017年11月14日。[31] Loshchilov I、Hutter F。SGDR：带热重启的随机梯度下降。arXiv预印本 arXiv:1608.03983。2016年8月13日。[32] Zhou Wang、A. C. Bovik、H. R. Sheikh 和 E. P. Simoncelli，“图像质量评估：从误差可见性到结构相似性”，载于《IEEE图像处理汇刊》，第13卷，第4期，第600-612页，2004年4月，doi：10.1109/TIP.2003.819861。[33] Zhou、Qingnan 和 Jacobson、Alec，Thingi10 10,000 3 2016，1605 04797K：一个三维打印模型数据集，arXiv预印本 arXiv：。。