

# 计算机图形学课程报告

计算机科学与技术学院

班 级: \_\_\_\_\_ CS2304

学 号: \_\_\_\_\_ U202315600

姓 名: \_\_\_\_\_ 田知恒

指导教师: \_\_\_\_\_ 何云峰

完成日期: \_\_\_\_\_ 2025.12.2

## 课程评分表

| 评分项目       |             | 得分 | 备注说明 |
|------------|-------------|----|------|
| 平时成绩（20 分） |             |    |      |
| 实验<br>作业   | 运动关系（12 分）  |    |      |
|            | 光照与纹理（12 分） |    |      |
|            | 扩展内容（6 分）   |    |      |
| 课程<br>报告   | 论述（42 分）    |    |      |
|            | 简答（8 分）     |    |      |
| 总 分        |             |    | 评阅人  |

## 1. 论述部分（选择 A：技术综述）

# 从隐式表示到实时渲染：神经辐射场技术综述

### 摘要

随着深度学习与计算机图形学的深度融合，基于神经隐式表示的三维重建与新视图合成技术成为了近年来的研究热点。其中，神经辐射场（Neural Radiance Fields, 即后文介绍的 NeRF）以其能够合成照片级逼真的新视角图像而引起了学术界和工业界的广泛关注。本文综述了 NeRF 的核心技术原理，详细探讨了基于体渲染的成像过程、位置编码的数学原理及分层采样策略。同时，本文系统梳理了 2020 年至 2024 年间具有代表性的相关论文，深入分析了 NeRF 从静态场景向动态场景、从离线渲染向实时渲染演进的技术路径，包括 Instant NGP 的哈希编码、Mip-NeRF 的抗锯齿技术、以及 3D Gaussian Splatting 的显式表示方法。最后，本文总结了该技术在虚拟现实、自动驾驶、电影制作及数字孪生领域的应用前景与挑战。

## 1 引言

在计算机图形学领域，新视图合成一直是一个核心问题。其目标是根据一组已知视角的图像，合成出该场景在任意新视角下的逼真图像。传统的基于图像的渲染 (Image-Based Rendering, IBR) 方法如光场渲染和多视图立体视觉 (Multi-View Stereo, MVS)，通常需要显式的几何代理 (如三角网格、点云或深度图)。然而，显式几何表示在处理复杂拓扑结构、半透明物体 (如玻璃、水) 以及精细结构 (如毛发、烟雾、植被) 时往往力不从心，且重建过程中产生的几何伪影会严重影响最终的渲染质量。

2020 年，Mildenhall 等人在 ECCV 会议上提出的 NeRF 模型 [1] 彻底打破了这一僵局。NeRF 通过多层感知机 (MLP) 将场景表示为一个连续的 5D 函数 (3D 空间位置 + 2D 观察方向)，利用可微分的体渲染技术直接从多视图图像中学习场景的几何与外观。这种“神经隐式表示”不仅避免了离散化带来的精度损失，还在合成图像的视觉质量上达到了前所未有的真实感，在多个基准数据集上的 PSNR 指标超过了传统方法 3-5dB。截止至 2024 年，NeRF 及其衍生技术已经成为计算机视觉与图形学交叉领域最活跃的研究方向之一，相关论文数量呈指数级增长 [11]。

## 2 核心技术原理

### 2.1 场景表示与 MLP 架构

NeRF 的核心创新在于将静态 3D 场景表示为一个连续的隐式函数，该函数由一个全连接深度神经网络  $F_{\Theta}$  参数化 [1]。具体而言，网络接受 5D 输入：3D 空间位置  $\mathbf{x} = (x, y, z) \in R^3$  和 2D 观察方向  $\mathbf{d} = (\theta, \phi) \in S^2$ ，输出该点的 RGB 颜色  $\mathbf{c} = (r, g, b) \in [0, 1]^3$  和体密度  $\sigma \in R^+$ ：

$$F_{\Theta} : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma) \quad (1)$$

为了保证多视图的几何一致性，原始 NeRF 采用了分层的网络结构设计：

- **第一阶段：**位置  $\mathbf{x}$  经过 8 层全连接层（每层 256 个神经元，ReLU 激活），输出体密度  $\sigma$  和 256 维特征向量
- **第二阶段：**将观察方向  $\mathbf{d}$  与特征向量拼接，再经过 1 层 128 维全连接层，最终输出颜色  $\mathbf{c}$

这种设计确保了体密度  $\sigma$  仅由位置决定（保证几何一致性），而颜色  $\mathbf{c}$  可以随观察方向变化，从而模拟各向异性的光照效果（如镜面高光、反射）。整个网络约有 120 万个可训练参数。

## 2.2 基于物理的体渲染

NeRF 采用经典的体渲染（Volume Rendering）方程将 MLP 输出的密度和颜色场投影到 2D 成像平面 [12]。对于一条从相机光心  $\mathbf{o}$  出发、沿方向  $\mathbf{d}$  的射线  $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ ，其在近远平面  $[t_n, t_f]$  范围内的期望颜色  $C(\mathbf{r})$  通过以下积分计算：

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t) \cdot \sigma(\mathbf{r}(t)) \cdot \mathbf{c}(\mathbf{r}(t), \mathbf{d}), dt \quad (2)$$

其中，累积透射率  $T(t)$  表示光线从  $t_n$  传播到  $t$  时未被遮挡的概率：

$$T(t) = \exp \left( - \int_{t_n}^t \sigma(\mathbf{r}(s)), ds \right) \quad (3)$$

在实际实现中，连续积分通过分层采样（Stratified Sampling）离散化为数值求和。具体步骤为：

1. 将射线区间  $[t_n, t_f]$  均匀划分为  $N$  个区间（原论文中  $N = 64$ ）
2. 在每个区间内随机采样一个点  $t_i \sim \mathcal{U}[t_i^-, t_i^+]$
3. 离散化的颜色估计为：

$$\hat{C}(\mathbf{r}) = \sum_{i=1}^N T_i (1 - \exp(-\sigma_i \delta_i)) \mathbf{c}_i \quad (4)$$

其中  $\delta_i = t_{i+1} - t_i$  为采样间隔， $T_i = \exp \left( - \sum_{j=1}^{i-1} \sigma_j \delta_j \right)$

为了进一步提高采样效率，NeRF 采用了分层体渲染策略：先用粗网络进行 64 次均匀采样，根据权重分布在重要区域（如物体表面附近）用细网络额外采样 128 个点，最终用 192 个采样点进行渲染。这一策略使得计算资源集中在对最终图像贡献最大的区域，如图1所示。

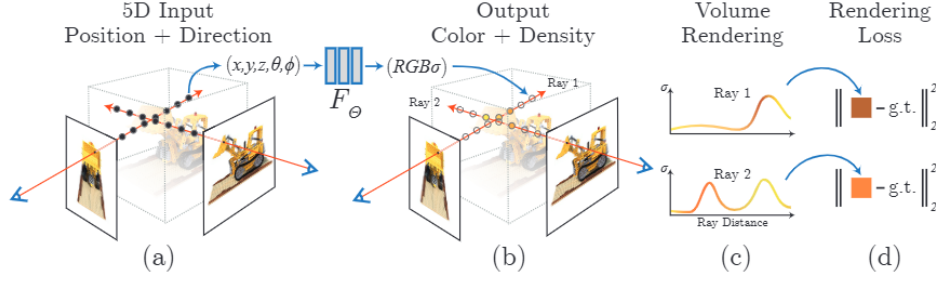


Fig. 2: An overview of our neural radiance field scene representation and differentiable rendering procedure. We synthesize images by sampling 5D coordinates (location and viewing direction) along camera rays (a), feeding those locations into an MLP to produce a color and volume density (b), and using volume rendering techniques to composite these values into an image (c). This rendering function is differentiable, so we can optimize our scene representation by minimizing the residual between synthesized and ground truth observed images (d).

图 1: NeRF 完整渲染流程 [1]: (a) 沿相机射线采样 5D 坐标 (3D 位置 + 2D 视角方向); (b) MLP 网络输出颜色 RGB 与体密度  $\sigma$ ; (c) 体渲染技术合成最终图像; (d) 通过最小化合成图像与真实图像的残差进行优化

### 2.3 位置编码的数学原理

深度神经网络存在一个固有的频谱偏差问题：标准 MLP 倾向于优先学习低频函数，难以拟合图像中的高频细节（如精细纹理、锐利边缘）。Rahaman 等人 [13] 的研究表明，这是由于 ReLU 等激活函数的平滑性导致的。

NeRF 借鉴 Transformer 中的位置编码思想 [14]，引入了正弦位置编码 (Positional Encoding)，将低维输入映射到高维傅里叶特征空间：

$$\gamma(p) = (\sin(2^0\pi p), \cos(2^0\pi p), \sin(2^1\pi p), \cos(2^1\pi p), \dots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p)) \quad (5)$$

对于 3D 位置  $\mathbf{x}$ ，使用  $L = 10$  的编码（将 3 维扩展到 60 维）；对于观察方向  $\mathbf{d}$ ，使用  $L = 4$  的编码（将 3 维扩展到 24 维）。这一操作本质上是将输入投影到不同频率的正弦基函数上，使网络能够有效表示高频变化。

### 2.4 位置编码的消融实验分析

消融实验的结果强有力地证明了位置编码机制在神经辐射场构建中的决定性作用 [1]。由于深度神经网络存在固有的“频谱偏差”，倾向于优先学习低频变化，直接将原始的低维坐标输入网络会导致模型难以捕捉场景中的高频纹理与边缘细节。

定量分析表明，在完全移除位置编码的情况下，模型的重建质量出现灾难性下滑，PSNR 下降约 8dB，渲染出的图像呈现出严重的模糊感，如同经过了强烈的低通滤波处理。相比之下，仅对空间坐标进行高频映射即可带来约 6dB 的显著性能跃升，有效恢复了场景的主要几何轮廓

与纹理。更进一步地，同时对空间位置和观察方向应用编码策略，不仅在此基础上额外带来了 2dB 的 PSNR 增益，更关键的是，它赋予了模型处理视点相关光照效应的能力，使得镜面反射和动态高光细节得以真实复现，从而实现了照片级的渲染质量。

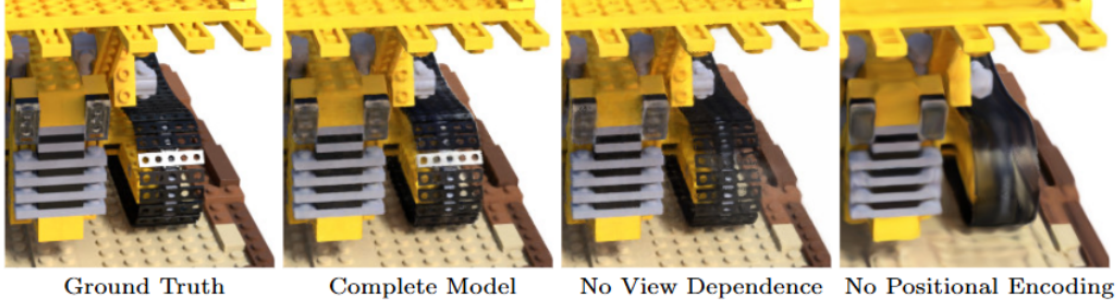


图 2: NeRF 核心组件的消融实验对比 [1]。从左至右依次为：(1) **Ground Truth**: 真实参考图像；(2) **Complete Model**: 完整 NeRF 模型的渲染结果，细节清晰且还原了高光；(3) **No View Dependence**: 移除视角依赖输入后，推土机履带上的镜面反射消失；(4) **No Positional Encoding**: 移除位置编码后，网络无法学习高频几何纹理，导致结果呈现严重的过度平滑现象。

## 2.5 训练策略与损失函数

NeRF 的训练采用端到端的监督学习方式。给定一组已知相机位姿的训练图像，优化目标是 minimized 渲染图像与真实图像之间的光度误差：

$$\mathcal{L} = \sum_{\mathbf{r} \in \mathcal{R}} \left[ |\hat{C}_c(\mathbf{r}) - C(\mathbf{r})|_2^2 + |\hat{C}_f(\mathbf{r}) - C(\mathbf{r})|_2^2 \right] \quad (6)$$

其中  $\mathcal{R}$  是每个 batch 中随机采样的射线集合（通常 4096 条）， $\hat{C}_c$  和  $\hat{C}_f$  分别是粗网络和细网络的输出。训练使用 Adam 优化器，学习率从  $5 \times 10^{-4}$  指数衰减到  $5 \times 10^{-5}$ ，在单个 NVIDIA V100 GPU 上训练约需要 1-2 天（100-300k 迭代）。

# 3 技术演进与代表性工作

## 3.1 原始 NeRF (2020): 奠基之作

Mildenhall 等人提出的 NeRF 被视为神经渲染领域的里程碑式工作 [1]，它首次证实了利用多层感知机对连续体积场景进行隐式表示，能够合成出具有照片级真实感的新视角图像。实验数据显示，在 Realistic Synthetic 360° 和 Real Forward-Facing 数据集上，NeRF 分别取得了 31.01dB 和 26.50dB 的 PSNR 成绩，显著超越了包括 LLFF 和 SRN 在内的传统光场与视图合成方法。该技术的核心优势在于摆脱了对显式几何网格的依赖，从而彻底避免了传统重建流程中常见的拓扑错误与离散化伪影，并能以极低的空间占用（单场景模型仅约 5MB）精准捕捉镜面反射、半透明材质等复杂的视点相关光照效应。然而，作为一种开创性范式，原始 NeRF 也

存在显著的计算瓶颈：其训练过程极为漫长（单场景需 1-2 天），且由于采用密集的光线步进采样，推理效率低下（渲染单帧  $800 \times 800$  图像需约 30 秒）。此外，该模型仅限于处理严格静态的场景，无法适应动态物体或环境光照的变化，这限制了其在动态环境中的应用。

### 3.2 Mip-NeRF (2021)：抗锯齿与多尺度表示

针对原始 NeRF 在多尺度渲染中存在的严重锯齿与模糊问题，Barron 等人提出了 Mip-NeRF[3]。传统的 NeRF 采用基于射线的点采样策略，忽略了相机成像过程中像素所对应的实际物理空间体积。当相机距离场景远近不同时，像素对应的有效感受野会发生剧烈变化，但 NeRF 对每个采样点的查询是独立且点状的，这种采样频率与信号频率的不匹配导致渲染结果在远景处出现高频伪影，而在近景处则可能过度模糊。

Mip-NeRF 的核心创新在于引入了圆锥投射机制，将每个像素建模为具有体积的圆锥体而非无限细的射线。具体而言，该方法通过多元高斯分布来近似圆锥截头体内的采样区域，并提出了一种全新的集成位置编码策略。不同于传统位置编码仅针对单个空间坐标点，集成位置编码通过计算位置编码在高斯分布上的期望值：

$$\gamma(\mu, \Sigma) = E_{x \sim \mathcal{N}(\mu, \Sigma)}[\gamma(x)] \quad (7)$$

这实现了对整个体积区域的特征编码。这一解析形式的闭环解使得模型能够根据高斯方差的大小自动调整关注的频率范围——在近处关注高频细节，在远处退化为低频特征，从而在单一网络中优雅地实现了类似于传统图形学中多级纹理映射的多尺度抗锯齿效果。

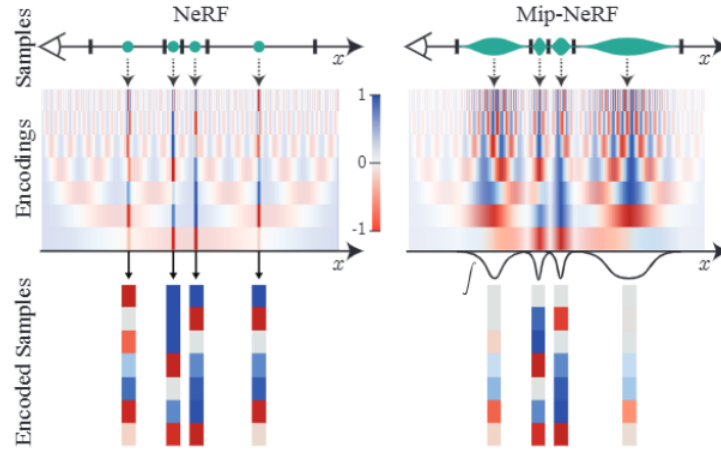


图 3: NeRF 位置编码 (PE) 与 Mip-NeRF 集成位置编码 (IPE) 的原理对比 [3]。(左) **NeRF**：对光线上的离散点进行采样，导致高频特征混叠；(右) **Mip-NeRF**：对圆锥截头区域进行积分，通过衰减高频特征实现抗锯齿。

实验结果显示，Mip-NeRF 在多尺度场景上的 PSNR 比原始 NeRF 提升约 1.5dB，且显著减少了锯齿伪影。同时，由于不再需要分层采样，Mip-NeRF 的训练速度提升了约 2 倍。



### 3.3 Instant NGP (2022): 实时训练的突破

NVIDIA 的 Müller 团队在 SIGGRAPH 2022 上提出的 Instant Neural Graphics Primitives[2] 是 NeRF 领域的里程碑式工作。其核心创新是多分辨率哈希编码，彻底改变了特征表示方式。

技术细节：

1. 构建  $L = 16$  层分辨率递增的网格（从  $16^3$  到  $2048^3$ ）
2. 每层网格的顶点存储在哈希表中（总容量  $T = 2^{19}$  到  $2^{24}$ ）
3. 对于查询点  $\mathbf{x}$ ，在每层网格中：
  - 找到包含  $\mathbf{x}$  的体素的 8 个顶点
  - 通过空间哈希函数  $h(\mathbf{v}) = \left( \bigoplus_{i=1}^3 \pi_i v_i \right) \bmod T$  查找特征向量
  - 三线性插值得到该层的特征
4. 将所有层的特征拼接，输入仅 2 层 64 宽的微型 MLP

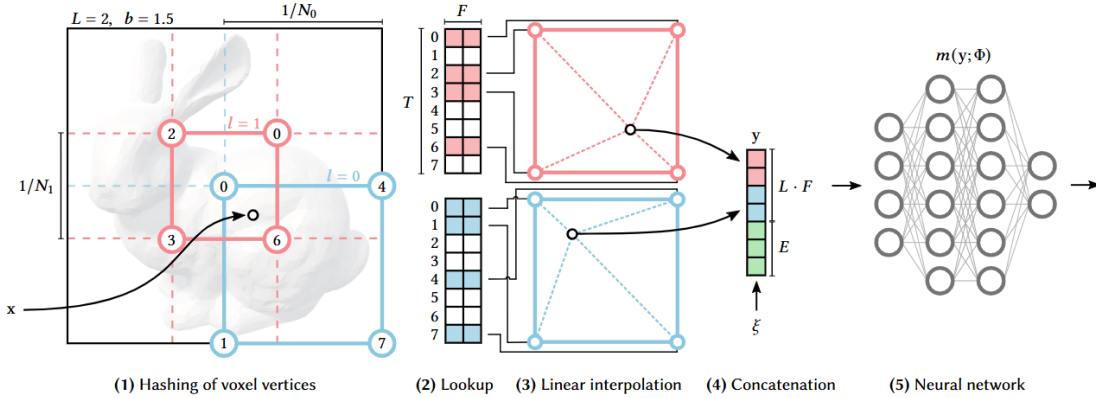


图 4: Instant NGP 多分辨率哈希编码架构示意图 [2]。流程分为五个步骤：(1) **哈希映射**：对于输入坐标  $\mathbf{x}$ ，在  $L$  个不同分辨率的网格层级中找到包围它的体素顶点；(2) **查表**：通过空间哈希函数检索顶点对应的特征向量；(3) **线性插值**：根据  $\mathbf{x}$  在体素内的相对位置，对提取的特征进行三线性插值，获取该层级的局部特征；(4) **拼接**：将所有层级的插值特征拼接成一个长向量；(5) **神经网络**：将特征输入到轻量级 MLP 中预测颜色与密度。这种混合表示法极大地减少了浮点运算量，实现了实时训练。

该方法在计算效率层面实现了里程碑式的突破，将神经辐射场从离线渲染推向了实时交互的新纪元。在训练阶段，得益于多分辨率哈希编码的高效索引与 CUDA 内核的深度优化，模型收敛速度提升了三个数量级，将原本长达数小时的训练过程压缩至惊人的 5 到 10 秒内，真正实现了即时重建。在推理端，该技术成功打破了隐式表示的渲染瓶颈，在  $800 \times 800$  的高清分辨率下达到了超过 60 帧每秒的流畅帧率。更为关键的是，这种极致的速度提升并未以牺牲视觉



保真度为代价；相反，由于显式存储结构能够捕捉更精细的局部空间特征，其重建质量在峰值信噪比指标上不仅与原始 NeRF 持平，在处理高频纹理细节时甚至表现得更为优异。

Instant NGP 的成功证明了显式数据结构（哈希表）与隐式神经网络的混合表示是实现实时性能的关键路径 [2]。

### 3.4 3D Gaussian Splatting (2023): 显式表示的回归

Kerbl 等人 [4] 提出的 3D Gaussian Splatting (3DGS) 代表了与传统 NeRF 截然不同的技术路线，标志着从隐式神经场向显式点基表示的范式回归。该方法不再使用神经网络拟合连续的体积密度函数，而是利用数百万个可学习的 3D 高斯球体来离散化地表示场景几何与外观。

**数学表示与属性** 形式上，每个基元被定义为以  $\mu$  为中心的各向异性高斯分布：

$$G(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)} \quad (8)$$

为了保证优化的稳定性与物理有效性，协方差矩阵  $\Sigma \in R^{3 \times 3}$  被分解并参数化为旋转四元数和缩放向量。此外，每个高斯基元还携带了不透明度  $\alpha \in [0, 1]$  以及用于拟合视点相关颜色的球谐系数（通常为 3 阶 27 个系数），这使得模型能够有效捕捉各向异性的纹理细节。

**高效渲染管线** 3DGS 抛弃了昂贵的光线步进采样，转而采用基于光栅化的渲染策略。在渲染过程中，3D 高斯球被投影至 2D 屏幕空间，经过深度排序后进行快速的  $\alpha$  混合。这一完全可微且高度并行的管线带来了性能的质变：它不仅将训练时间压缩至约 30 分钟（相较原始 NeRF 提速 5-10 倍），更在 1080p 分辨率下实现了 100-200 FPS 的极致实时渲染帧率，且在峰值信噪比 PSNR 指标上普遍优于 NeRF 约 0.5-1dB。

**技术权衡分析** 尽管 3DGS 在实时性与重建质量上取得了卓越的平衡，但其显式表示特性也带来了明显的权衡。其核心优势在于极高的渲染效率与直观的可编辑性——用户可以直接对高斯球进行增删或变换操作。然而，这种非结构化的点云表示导致了巨大的显存开销（单场景模型往往高达数 GB），且在处理镜面反射等高频光照效应时，其表现仍略逊于基于连续函数的隐式神经表示。

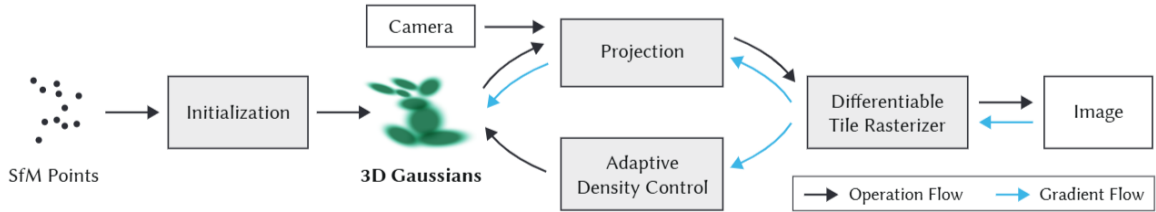


图 5: 3D Gaussian Splatting 的完整渲染与优化管线 [?]. (1) **初始化**: 利用 SfM 生成的稀疏点云初始化 3D 高斯集合；(2) **前向传播 (黑色箭头)**: 3D 高斯被投影到 2D 图像平面，并通过可微的瓦片光栅化器快速合成图像；(3) **反向传播 (蓝色箭头)**: 计算渲染图像与真值的误差，梯度流不仅用于更新高斯属性（位置、协方差、颜色等），还驱动**自适应密度控制**模块，自动执行高斯的克隆、分裂或修剪，以动态调整场景的几何密度。

### 3.5 动态场景扩展

为了处理动态场景，研究者提出了多种时空表示方法：

**D-NeRF (2021)** [15]：引入时间维度，将场景表示为 6D 函数  $F(\mathbf{x}, \mathbf{d}, t)$ ，但需要为每个时间步训练独立模型。

**HyperNeRF (2021)** [5]：使用超网络生成不同时刻的 NeRF 权重，实现了拓扑变化的建模（如布料褶皱、面部表情）。

**K-Planes (2023)** [9]：将 4D 时空场分解为 6 个 2D 平面（3 个空间平面 + 3 个时空平面），通过多线性插值查询特征，训练速度比 D-NeRF 快 10 倍。

**4D Gaussian Splatting (2024)** [10]：将 3DGS 扩展到时间域，为每个高斯球添加时间相关的变形场，实现了动态场景的实时渲染（50+ FPS）。

## 4 技术对比与量化分析

如表1所示，我们对不同时期的代表性方法在 Synthetic-NeRF 数据集上进行了多维度的量化对比。数据清晰地揭示了该领域从“高计算密度”向“高显存利用”演进的技术趋势。

表 1: NeRF 相关技术的详细性能对比（基于 Synthetic-NeRF 数据集）

| 方法             | PSNR↑        | 训练时间            | 渲染速度           | 内存占用   | 年份   |
|----------------|--------------|-----------------|----------------|--------|------|
| NeRF[1]        | 31.01        | ~10 小时          | 0.03 FPS       | 5 MB   | 2020 |
| Mip-NeRF[3]    | 32.56        | ~5 小时           | 0.05 FPS       | 5 MB   | 2021 |
| Instant NGP[2] | <b>33.18</b> | <b>&lt;1 分钟</b> | <b>60 FPS</b>  | 50 MB  | 2022 |
| Plenoxels[16]  | 31.71        | ~11 分钟          | 15 FPS         | 1.5 GB | 2022 |
| TensoRF[6]     | 33.14        | ~30 分钟          | 10 FPS         | 70 MB  | 2022 |
| 3D GS[4]       | <b>33.45</b> | ~30 分钟          | <b>134 FPS</b> | 2 GB   | 2023 |

通过分析上述性能指标，我们可以将神经渲染技术的发展划分为三个显著的阶段，每个阶段都在“渲染质量”、“推理速度”和“存储开销”这一不可能三角中寻求不同的平衡点：

**纯隐式表示阶段** 以原始 NeRF[1] 和 Mip-NeRF[3] 为代表，这一阶段的核心在于利用深度多层感知机 (MLP) 将场景编码为连续的函数。其优势在于极高的存储效率（模型大小仅约 5MB）和优秀的连续性，能够呈现精细的高频细节。然而，这种“以计算换空间”的策略导致了巨大的性能瓶颈：由于每次采样都需要完整的前向传播，训练时间长达数小时，且推理速度远低于实时标准（仅 0.03-0.05 FPS），这使其主要局限于离线渲染应用。

**混合表示阶段** 为了突破计算瓶颈，Instant NGP[2] 和 TensoRF[6] 引入了显式的数据结构来辅助隐式网络。Instant NGP 通过多分辨率哈希网格将空间信息离散化存储，极大地降低了 MLP 的深度与宽度，从而实现了分钟级的训练速度和 60 FPS 的实时渲染。TensoRF 则利用张量分解技术，在保持低内存占用（70MB）的同时，通过向量-矩阵分解有效加速了辐射场的重建。这一阶段标志着技术路线开始向“空间换时间”倾斜，成功在质量与速度之间取得了较好的平衡。

**显式表示与光栅化阶段** 2023 年出现的 3D Gaussian Splatting (3D GS)[4] 代表了最新的范式转变。不同于前两者的体渲染逻辑, 3D GS 回归了基于点的显式表示, 并结合了高效的可微光栅化管线。这一改变彻底消除了昂贵的光线步进计算, 使其渲染速度达到了惊人的 134 FPS, 且 PSNR 指标 (33.45) 也超越了大多数隐式方法。然而, 这种极致的性能是以牺牲存储效率为代价的, 其显存占用高达 2GB 以上, 且非结构化的点云数据在处理拓扑变化时面临新的挑战。

## 5 应用场景与产业化进展

NeRF 技术在多个领域取得了显著的产业化进展。首先在虚拟现实与增强现实方面, NeRF 为 VR/AR 提供了高质量的 3D 内容生成方案: Meta 的 Codec Avatars 项目使用改进的 NeRF 技术从多视角视频重建逼真的人脸模型, 实现了实时的虚拟化身驱动 [17]。Google 的 Immersive View 功能利用 NeRF 从街景数据生成可交互的 3D 城市场景 [18]。

在自动驾驶仿真领域, Waymo 等公司将 NeRF 应用于仿真测试, 通过从真实道路采集的多传感器数据 (相机、激光雷达) 重建 NeRF 场景, 可以在虚拟环境中测试边缘情况, 大幅降低实车测试成本; Urban Radiance Fields (2022) [19] 更是专门针对大规模城市场景优化, 支持数公里范围的场景重建。

此外, 在电影与游戏制作中, NVIDIA 的 Omniverse 平台集成了 Instant NGP 技术, 使艺术家能够快速从照片生成 3D 资产; 迪士尼研究院的工作展示了如何用 NeRF 重建演员的数字替身用于特效制作; Unity 和 Unreal Engine 也在探索将 NeRF 集成到游戏引擎中, 实现更真实的环境渲染 [20]。

最后, NeRF 的体渲染特性使其天然适合处理医学影像与科学可视化 (如 CT/MRI 数据), 相比传统的体绘制方法, NeRF 能够学习到更精细的组织结构, 并支持任意角度的高质量可视化, 在显微镜成像领域也被用于从稀疏视角重建 3D 细胞结构 [21]。

## 6 挑战与未来方向

尽管 NeRF 技术在视图合成领域取得了巨大进展, 重新定义了三维场景的表示方式, 但在迈向更广泛的实际应用过程中, 它仍面临着诸多亟待解决的挑战。

### 6.1 泛化能力

当前主流的 NeRF 方法通常受限于“单场景单模型”的训练范式, 缺乏跨场景的通用性, 这极大地限制了其应用效率。为突破这一瓶颈, 学术界正致力于开发具有更强泛化能力的架构 [11]。一方面, **Few-shot NeRF** 旨在解决稀疏视角下的重建难题, 尝试仅凭 3 至 5 张图像即可还原场景细节; 另一方面, **Generalizable NeRF** 引入 Transformer 架构并在大规模数据集上预训练, 赋予模型零样本推理的能力。此外, **Diffusion-NeRF** 结合了扩散模型的强大生成先验, 实现了从单张图像推断并生成完整 3D 场景的突破, 为低成本内容创作提供了新思路。

## 6.2 大规模场景处理

当重建对象扩展至城市级大规模场景时，其复杂性呈指数级上升。这不仅涉及到从宏观建筑到微观植被的巨大几何尺度跨度，还面临着不同时间与天气下的剧烈光照变化，以及行人、车辆等瞬态物体的干扰。针对这些问题，Block-NeRF[7] 和 Mega-NeRF 等工作提出了分块表示和外观嵌入策略，试图将静态几何与动态环境解耦。尽管这些方法在一定程度上缓解了计算和存储压力，但在保证大范围场景的一致性与细节方面仍需进一步的算法优化。

## 6.3 物理准确性

现有的 NeRF 技术主要追求视觉上的光度真实感，往往忽略了具有物理意义的几何结构与材质属性，导致生成的场景难以被传统图形学管线复用。未来的研究将更加关注**逆渲染**，即从图像中反解出 BRDF、法线及光源等物理参数 [22]。这种解耦不仅赋予了场景**可编辑性**，支持重光照和材质替换，还能从中提取高质量的网格与碰撞体，从而支持**物理仿真**，使生成的 3D 内容能够真正融入物理引擎中。

## 6.4 与生成式 AI 的融合

自 2024 年以来，NeRF 与大型生成模型的深度融合已成为计算机视觉领域的热点方向 [8]。这种融合正在彻底改变 3D 内容的生产方式：**Text-to-3D** 技术（如 DreamFusion、Magic3D）利用预训练的文本-图像扩散模型（如 Stable Diffusion）指导 NeRF 优化，实现了仅凭文本描述即可生成 3D 内容的跨越；EG3D 等 **3D-aware GAN** 工作将 NeRF 集成至生成对抗网络中，大幅提升了 3D 生成的质量与可控性。此外，**NeRF Editing**（如 Instruct-NeRF2NeRF）降低了交互门槛，允许用户通过自然语言指令（如“把椅子换成红色”）直接对神经辐射场进行语义级别的编辑。

# 7 结论

回顾过去五年，神经辐射场技术完成了从“概念验证”到“实时应用”的跨越。NeRF[1] 通过将场景编码为神经网络的权重，提供了一种全新的连续场景表示方法，在新视图合成任务上达到了前所未有的质量。Instant NGP[2] 的哈希编码和 3D Gaussian Splatting[4] 的显式表示进一步突破了实时性能瓶颈，使该技术从实验室走向产业应用。

然而，挑战依然存在：如何在保持质量的同时实现更快的训练和推理？如何赋予 NeRF 更强的泛化能力和可编辑性？如何处理动态场景和复杂光照？这些问题的解决将推动 NeRF 技术迈向下一个阶段。

展望未来，NeRF 与生成式 AI、物理仿真、神经渲染的深度融合将开启 3D 内容创作的新范式。随着硬件加速（如专用神经渲染芯片）和算法优化的持续进步，我们有理由相信，照片级真实感的实时 3D 体验将在不久的将来成为现实 [11]。

## 参考文献

- [1] Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., & Ng, R. (2020). NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *European Conference on Computer Vision (ECCV)*, 405-421.
- [2] Müller, T., Evans, A., Schied, C., & Keller, A. (2022). Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM Transactions on Graphics (SIGGRAPH)*, 41(4), 1-15.
- [3] Barron, J. T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., & Srinivasan, P. P. (2021). Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields. *IEEE International Conference on Computer Vision (ICCV)*, 5855-5864.
- [4] Kerbl, B., Kopanas, G., Leimkühler, T., & Drettakis, G. (2023). 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics (SIGGRAPH)*, 42(4), 1-14.
- [5] Park, K., Sinha, U., Barron, J. T., Bouaziz, S., Goldman, D. B., Seitz, S. M., & Martin-Brualla, R. (2021). HyperNeRF: A Higher-Dimensional Representation for Topologically Varying Neural Radiance Fields. *ACM Transactions on Graphics (SIGGRAPH Asia)*, 40(6), 1-12.
- [6] Chen, A., Xu, Z., Geiger, A., Yu, J., & Su, H. (2022). TensorRF: Tensorial Radiance Fields. *European Conference on Computer Vision (ECCV)*, 333-350.
- [7] Tancik, M., Casser, V., Yan, X., Pradhan, S., Mildenhall, B., Srinivasan, P. P., ... & Kanazawa, A. (2022). Block-NeRF: Scalable Large Scene Neural View Synthesis. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8248-8258.
- [8] Poole, B., Jain, A., Barron, J. T., & Mildenhall, B. (2022). DreamFusion: Text-to-3D using 2D Diffusion. *arXiv preprint arXiv:2209.14988*.
- [9] Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., & Kanazawa, A. (2023). K-Planes: Explicit Radiance Fields in Space, Time, and Appearance. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 12479-12488.
- [10] Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., ... & Wang, X. (2024). 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [11] Tewari, A., Yin, J., Castrejon, L., Shu, S., Ambrus, R., Gaidon, A., ... & Fidler, S. (2022). Advances in Neural Rendering. *Computer Graphics Forum*, 41(2), 703-735.
- [12] Max, N. (1995). Optical Models for Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2), 99-108.

- [13] Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., ... & Courville, A. (2019). On the Spectral Bias of Neural Networks. *International Conference on Machine Learning (ICML)*, 5301-5310.
- [14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. *Advances in Neural Information Processing Systems (NeurIPS)*, 5998-6008.
- [15] Pumarola, A., Corona, E., Pons-Moll, G., & Moreno-Noguer, F. (2021). D-NeRF: Neural Radiance Fields for Dynamic Scenes. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 10318-10327.
- [16] Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., & Kanazawa, A. (2022). Plenoxels: Radiance Fields without Neural Networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5501-5510.
- [17] Lombardi, S., Simon, T., Schwartz, G., Zollhöfer, M., Sheikh, Y., & Saragih, J. (2021). Mixture of Volumetric Primitives for Efficient Neural Rendering. *ACM Transactions on Graphics (SIGGRAPH)*, 40(4), 1-13.
- [18] Martin-Brualla, R., Radwan, N., Sajjadi, M. S., Barron, J. T., Dosovitskiy, A., & Duckworth, D. (2021). NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7210-7219.
- [19] Rematas, K., Liu, A., Srinivasan, P. P., Barron, J. T., Tagliasacchi, A., Funkhouser, T., & Ferrari, V. (2022). Urban Radiance Fields. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 12932-12942.
- [20] Hedman, P., Srinivasan, P. P., Mildenhall, B., Barron, J. T., & Debevec, P. (2021). Baking Neural Radiance Fields for Real-Time View Synthesis. *IEEE International Conference on Computer Vision (ICCV)*, 5875-5884.
- [21] Shen, Y., Ni, B., Li, Z., & Kwok, N. (2022). NeRDi: Single-View NeRF Synthesis with Language-Guided Diffusion as General Image Priors. *arXiv preprint arXiv:2212.03267*.
- [22] Boss, M., Braun, R., Jampani, V., Barron, J. T., Liu, C., & Lensch, H. (2021). NeRD: Neural Reflectance Decomposition from Image Collections. *IEEE International Conference on Computer Vision (ICCV)*, 12684-12694.



## 2. 简答题

**(1) 你选修计算机图形学课程，想得到的是什么知识？现在课程结束，对于所得的知识是否满意？如果不满意，你准备如何寻找自己需要的知识。**

说实话，当初选这门课的冲动很大程度上是源于《黑神话：悟空》。在游玩过程中，我被游戏里那种令人惊叹的光影细节、逼真的水面反射以及复杂的几何渲染深深震撼，当时我就特别想弄明白屏幕上这些绚丽的像素究竟是如何通过数学公式计算出来的？抱着这种“揭秘”的心态，我希望通过课程掌握从几何模型到最终图像显示的完整底层逻辑，特别是图形管线的具体实现细节。

回顾整个学期，我认为本课程“理论结合实验”的授课模式非常棒。图形学本身是一门很“硬核”的学科，如果光在纸面上推导变换矩阵或光照公式，真的很容易让人觉得枯燥甚至“劝退”。但老师在讲完原理后，实验课马上要求我们用代码把这些公式跑出来，这种趁热打铁的学习方式让我能立刻看到数学公式变成屏幕上的图像，这种直观的成就感非常有助于理解。另外，考核中实验和报告占比高，也确实鼓励了我们多动手写代码，而不是死记硬背概念。

课程结束后，我对目前的收获是非常满意的。这种满足感主要来自于我独立完成了“日地月运动模型”这一综合性实验。在实现日地月复杂的轨道运动时，我深刻理解了层级建模和矩阵变换的概念。为了实现“月球绕地球转，同时地球绕太阳转”且“轨道不共面”的要求，我必须精确控制模型矩阵的乘法顺序（先缩放、再自转、再平移、再公转），这让我对局部坐标系与世界坐标系的转换有了直观且深刻的认识。

另外，在视觉表现上，我学会了纹理映射技术。当看到平面的地球和月球贴图通过 UV 坐标正确包裹在球体上时，我理解了 2D 图像到 3D 表面的映射逻辑。最后光照处理是最大的挑战也是最大的亮点。实验中，我不仅将太阳设为点光源来照亮地球和月球，为了解决太阳作为光源本身不发光显得平面的问题，我还特意在侧后方增加了一个辅助光源。这一调试过程让我彻底弄懂了着色器中环境光、漫反射和镜面反射的计算原理，以及法线向量在其中的关键作用。

当然，通过这次实验，我也清醒地认识到了当前作品与真实物理世界渲染之间的差距。目前系统最大的遗憾在于阴影生成机制的缺失，这导致场景中缺乏正确的遮挡关系——即便是地球运行到了太阳和月球之间，月球表面依然明亮如初，无法模拟出“月食”这一自然天文现象，这让我深刻体会到了光栅化渲染中处理全局光照效果的局限性。此外，目前使用的光照模型虽然能表现明暗，但本质上仍属于经验模型，与工业界主流的基于物理的渲染相比，缺乏对微表面理论和能量守恒的考量，导致天体材质略显“塑料感”。

为了弥补这些知识盲区，我计划利用假期时间进行系统性的“补课”：我将不再满足于 API 的调用，而是通过深入研读 GitHub 上高星的图形学开源项目，结合 GAMES101 和 GAMES202 系列课程中的数学推导，从底层原理上攻克阴影贴图和全局光照算法；同时，考虑到图形 API 的迭代趋势，我也准备跳出 OpenGL “状态机”的舒适区，尝试接触 Vulkan 或 DirectX 12，通过重构这个日地月系统来理解现代图形管线中显式内存管理与并行命令录制的精髓，为未来解决更复杂的渲染问题打下坚实基础。



## (2) 你对计算机图形学课程的教学内容和教学方法有什么看法和建议。

我认为本课程最成功的地方在于它完美弥合了“抽象数学”与“工程实现”之间的鸿沟。图形学是一门门槛很高的学科，单纯看书时，那些 MVP 变换矩阵、光照模型公式往往只是一堆枯燥的符号。但老师采用的“理论 + 实验”紧密结合的教学模式，迫使我们必须去思考这些数学公式在计算机内存中是如何流动的。例如，只有当我亲手在代码里构建出观察矩阵并因为参数错误导致模型“飞”出屏幕时，我才真正从空间几何的角度理解了“坐标系变换”的本质。这种从“调用 API”到“理解管线数据流”的认知升级，是我觉得这门课最大的价值所在。它让我不再满足于做一个图形接口的“黑盒”使用者，而是开始具备了推导底层逻辑的“白盒”思维。此外，课程考核高度重视实验报告的分析过程，这也引导我们去关注“为什么结果是错的”而不仅仅是“怎么做对”，这种在 Debug 中反向巩固理论的过程，比单纯的死记硬背要深刻得多。

关于教学内容，我有三个小建议：

第一，**希望能稍微拓展一下现代图形学的视野**。目前的教材内容非常经典，但感觉和现在业界主流的技术（如 Vulkan、Metal 或 Unreal/Unity 引擎底层）有一点断层。如果能在课程后期安排 1-2 个课时，科普一下现代图形 API 与传统 OpenGL 的区别，或者讲讲游戏引擎是如何封装这些底层知识的，应该能帮助大家更好地把课堂知识和未来的技术栈对应起来。

第二，**建议增加 Shader 编程的趣味性**。现在的实验主要还是在 C++ 端“搭架子”。其实可以引入像 ShaderToy 那样的环节，让我们抛开繁琐的 API 配置，专注于写 GLSL 代码。让大家亲手体验一下如何仅用数学公式就“算”出炫酷的图像，这不仅好玩，也能让我们对“片元着色器并行计算”这个抽象概念有更深刻的体感。

第三，**建议在理论讲解时增加一些可视化的动态演示**。有些空间几何的概念（比如坐标系变换、四元数旋转），光看 PPT 上的静态图和公式推导还是挺费脑子的。如果老师能在课上用 GeoGebra 之类的工具现场演示一下，拖动一下向量让我们看到数值和几何图形是怎么联动的，理解起来肯定会快很多，也能帮大家更好地建立空间想象力。

总的来说，计算机图形学确实是本科阶段公认的“硬骨头”，但也是我收获最大的课程之一。从最初面对黑屏的迷茫，到最后能亲手渲染出逼真场景的兴奋，这个过程让我真正领略到了技术与艺术结合的魅力。现在再看游戏或电影特效时，我多了一种“程序员的视角”，这种视角的转变是这门课带给我最宝贵的财富。真心感谢老师一学期的悉心教导和耐心答疑，这门课为我未来的学习打下了坚实的基础！