

# Mip-NeRF：抗混叠的多尺度神经辐射场论文复现

李汭

## 摘要

三维重建是用相机拍摄真实世界的物体、场景，通过计算机视觉技术进行处理，从而得到物体的三维模型。而经辐射场 (Neural Radiance Field, NeRF) 作为一种具有隐式场景表示的新型视场合成技术，在计算机视觉领域引起了广泛的关注。作为一种新颖的视图合成和三维重建方法，NeRF 模型在机器人、城市地图、自主导航、虚拟现实/增强现实等领域都有广泛的应用。但是 NeRF 中还存在着诸多问题，本篇报告就主要针对 NeRF 再做渲染时，会出现模糊和锯齿的情况，而这种清空通常是由于同一个场景对应的多个图片的分辨率不一致而导致的。最简单的方式是对每个像素点多光线采样，但是这样对网络的开销是巨大的。于是 Mip-NeRF 就提出了用圆锥体来代替原先的光线线性采样，使得 Mip-NeRF 相对 NeRF 来说具有更快、更小、更准的优势。也让后续对 NeRF 的研究有了更好的效果，特别是在多尺度数据的处理上。鉴于对日后研究的帮助和需要，又由于其官方源码上传的是 JAX 版本的代码，并不利于日后研究中的工作，故本篇报告将对 Mip-NeRF 进行 Pytorch 版本复现，同时以求比官方得到更好的效果。

**关键词：** 三维重建；神经辐射场；视图合成；抗锯齿

## 1 引言

三维重建是指对三维物体建立适合计算机表示和处理的数学模型，是在计算机环境下对其进行处理、操作和分析其性质的基础，也是在计算机中建立表达客观世界的虚拟现实的关键技术。传统三维重建有深度图 (depth)、点云 (point cloud)、体素 (voxel)、网格 (mesh)。而 NeRF 则是一项利用多目图像重建三维场景的技术，通过基于隐式场表达来实现三维重建。该项目的作者来自于加州大学伯克利分校，Google 研究院，以及加州大学圣地亚哥分校。NeRF 使用一组多目图作为输入，通过优化一个潜在连续的体素场景方程来得到一个完整的三维场景<sup>[1]</sup>。该方法使用一个全连接深度网络来表示场景，使用的输入是一个单连通的 5D 坐标 (空间位置  $x, y, z$  以及观察视角  $\theta, \phi$ )，输出为一个体素场景，可以以任意视角查看，并通过体素渲染技术，生成需要视角的照片。该方法同样支持视频合成。NeRF 只在相机到对象距离固定的情况下可以生成表现优秀的结果，当相机拉近，拉远场景时会产生模糊和锯齿。产生的原因是采样频率低于真实原始信号的频率，为了解决这一问题，我们可以选择提高采样率或者粗暴去除高频分量（使用低通滤波器对边缘进行平滑）。但由于在 NeRF 中，是对每条光线进行采样并且对每条光线都要进行渲染，如果发射多条光线，提高了采样率，一定程度上可以解决模糊和锯齿问题，但这样的方法大大增加了计算量，效率太低，为此，mip-nerf 提出使用圆锥体取代光线的方案。Mip-NeRF 的解决方案和 NeRF 有一个本质不同，NeRF 渲染是要基于 ray 的，然而 Mip-NeRF 是基于 conical frustums (圆锥) 的，并且是抗混叠的。最终，Mip-NeRF 与 NeRF 相比具有更快、更小、更准的优势，更加适合处理多尺度的数据。并且 Mip-NeRF 在后续的 NeRF 领域研究中也被多次引用，包括可扩展的大场景神经视图合成 Block-NeRF 中<sup>[2]</sup>也使用了 Mip-NeRF 的方法对采样方式进行了改进。

## 2 相关工作

Mip-NeRF 的输入是一个三维高斯分布，表示辐射场应在其上积分的区域。<sup>[3]</sup>然后我们可以通过沿圆锥体每隔一段距离查询 mip-NeRF，使用近似于该像素对应的圆锥形截锥体的高斯分布来渲染像素。为了对 3D 位置及其周围的高斯分布进行编码，该论文提出了一种新的特征表示：集成位置编码（IPE, integrated positional encoding）。这是 NeRF 的位置编码（PE）的推广，它允许空间圆锥采样区域被紧凑地特征化，而不是空间中的单个点。在这种编码方式下，使得其锥体采样能被更好的编码，以提升体渲染的效果。

### 2.1 混叠现象

数据采集时，如果采样频率不满足奈奎斯特采样定理，可能会导致采样后的信号存在混叠，如图 1 所示。当采样频率设置不合理时，所定的取样频率若取样的频率太低，就会产生取样的结果和原来的样本不同的状况。若一样本的频谱是带限频谱，也就是在某一频率之外都为 0 的频谱，那么取样频率就必须要大于两倍的，才不至于使频谱产生交叠，也因此产生失真，即采样频率低于 2 倍的信号频率时，会导致原本的高频信号被采样成低频信号。如下图所示，红色信号是原始的高频信号，但是由于采样频率不满足采样定理的要求，导致实际采样点如图中实心点所示，将这些蓝色实际采样点连成曲线，可以明显地看出这是一个低频信号。

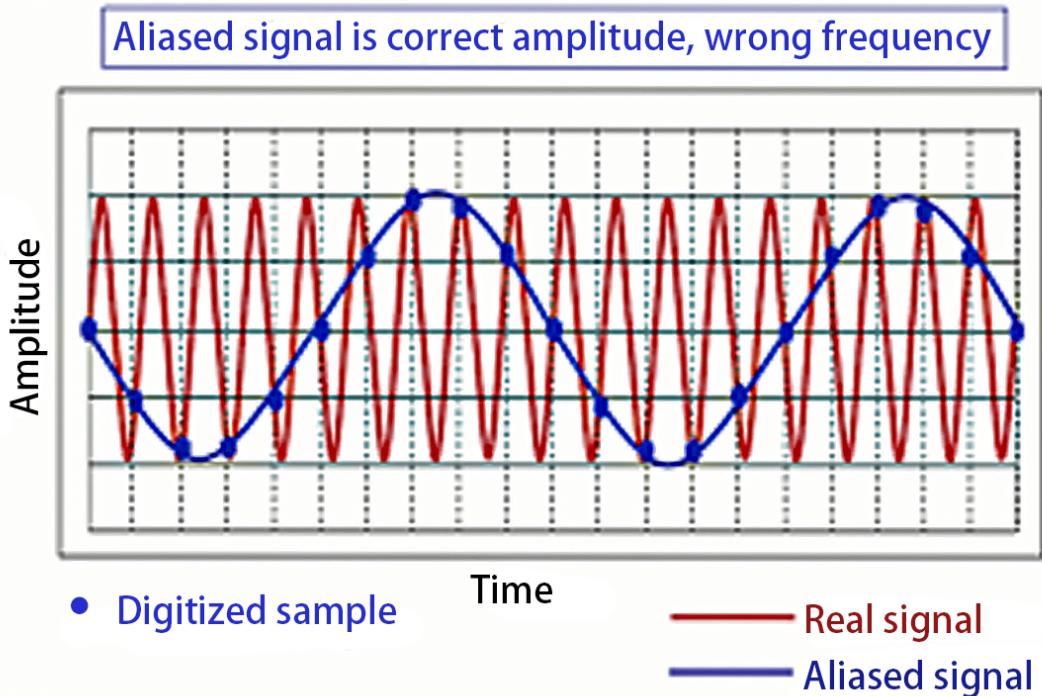


图 1: 信号混叠

NeRF 只在相机到对象距离固定的情况下可以生成表现优秀的结果，当相机拉近，拉远场景时会表现出明显的伪影。当训练图像以多种分辨率观察场景内容时，NeRF 的渲染在近景视图中显得过于模糊，并且在远景视图中包含混叠伪影。一个直接的解决方案是采用离线光线追踪中使用的策略：通过行进多条光线通过其足迹对每个像素进行超级采样。但这对于像 NeRF 这样的神经体积表示来说是非常昂贵的，它需要数百次 MLP 评估来渲染一条射线，并需要几个小时来重建一个场景。

## 2.2 抗混叠的多尺度神经辐射场

NeRF 对每一个像素只发射一条光线，如果发射多条光线，提高了采样率，一定程度上可以解决模糊和锯齿问题，但这样的方法大大增加了计算量，效率太低，为此抗混叠的多尺度神经辐射场提出使用圆锥体取代光线的方案。

该论文从用于防止计算机图形渲染管道中的混叠的 mipmapping 方法中获得灵感。ipmap 代表一组不同的离散下采样比例的信号（通常是图像或纹理贴图），并根据像素足迹到与该射线相交的几何体的投影，选择适当的比例用于射线。这种策略被称为预过滤，因为抗锯齿的计算负担从渲染时间（如在强力超级采样解决方案中）转移到預计算阶段 - 只需为给定纹理创建一次 mipmap，无论有多少渲染纹理的次数。

抗混叠的多尺度神经辐射场的输入是一个 3D 高斯分布，它表示应该对辐射场进行积分的区域。然后我们可以通过沿圆锥间隔查询 mip-NeRF 来渲染预过滤像素，使用高斯近似对应于像素的圆锥截头体。为了对 3D 位置及其周围的高斯区域进行编码，我们提出了一种新的特征表示：集成位置编码 (IPE)。这是 NeRF 位置编码 (PE) 的概括，它允许空间区域被紧凑地特征化，而不是空间中的单个点。

抗混叠的多尺度神经辐射场显着提高了神经辐射场的准确性，并且在以不同分辨率观察场景内容的情况下（即相机靠近和远离场景的设置），这种好处甚至更大。在提出的具有挑战性的多分辨率基准测试中，抗混叠的多尺度神经辐射场能够将相对于神经辐射场的错误率平均降低 60。抗混叠的多尺度神经辐射场的尺度感知结构还允许将 NeRF 用于分层采样的单独的“粗略”和“精细”的 MLP 合并到一个 MLP 中。因此，抗混叠的多尺度神经辐射场比 NeRF 稍快<sup>[3]</sup>，并且具有一半的参数。

## 3 本文方法

### 3.1 本文方法概述

和 NeRF 的工程流程（如图 2 所示）相似<sup>[1]</sup>，抗混叠的多尺度神经辐射场也是对光线追踪采样，但是与 NeRF 不同的是，抗混叠的多尺度神经辐射场的采样区域是一个圆台，而非单一光线，用多元高斯分布来近似这一圆台区域从而得到该采样区域的位置编码后的信息以及位姿信息。所以在抗混叠的多尺度神经辐射场中若需要 N 个采样点则需要采样 N+1 次，之后对采样圆台区域进行 IPE 得到位置编码和位姿信息，后续流程则基本与 NeRF 相同，但要注意的是，由于 NeRF 使用具有两个不同 MLP 的分层采样程序，一个“粗略”和一个“精细”。这在 NeRF 中是必要的，因为它的 PE 特性意味着它的 MLP 只能学习一个单一尺度的场景模型。但是抗混叠的多尺度神经辐射场的锥形投射和 IPE 特征允许我们将比例显式编码到我们的输入特征中，从而使 MLP 能够学习场景的多尺度表示因此 Mip-NeRF 使用带有参数  $\Theta$  的单个 MLP，在分层采样策略中重复查询。

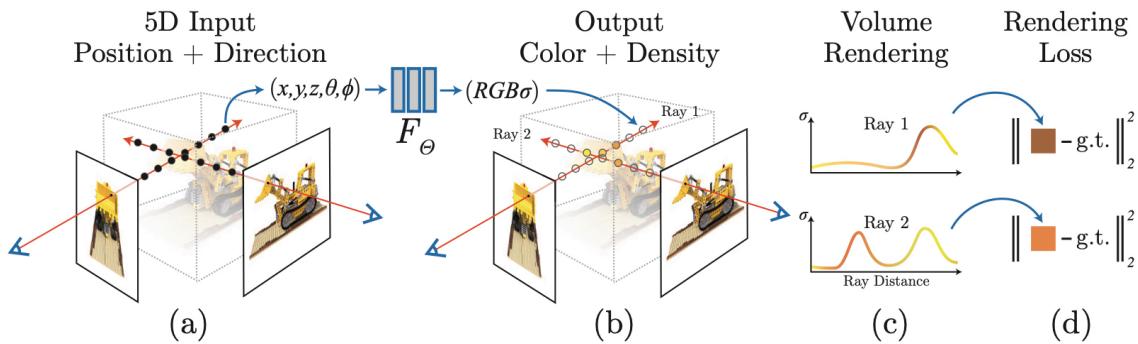


图 2: NeRF 渲染过程概述

### 3.2 锥体采样

MipNeRF 通过从每个像素投射一个锥体来改善这个问题<sup>[4]</sup>。Mip-NeRF 不是沿着每条射线执行点采样，而是将被投射的锥体分成一系列圆锥台（垂直于其轴切割的锥体），如图 3 所示。并且不是从空间中的无限小点构造位置编码 (PE) 特征，而是构造每个圆锥体所覆盖体积的集成位置编码 (IPE) 表示。这些变化允许 MLP 推断每个圆锥台的大小和形状，而不仅仅是它的质心。

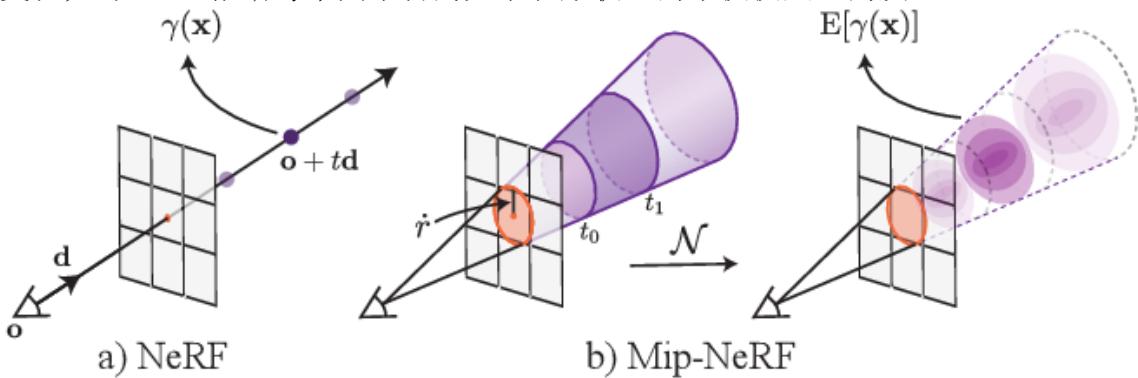


图 3: 锥体采样

图 4 显示了 NeRF 对尺度的不敏感性和 mip-NeRF 对这个问题的解决方案导致的歧义。这种使用圆锥台和 IPE 特征也允许我们将 NeRF 的两个独立的“粗”和“细”MLP 减少到一个单一的多尺度 MLP，提高训练和评估速度，并将模型大小减少一半。NeRF 的工作原理是沿着每个像素的光线提取点采样位置编码特征（此处显示为点）。这些点采样特征忽略了每条射线观察到的体积的形状和大小，因此两个不同的相机以不同的比例对同一位置进行成像可能会产生相同的模糊点采样特征，从而显着降低 NeRF 的性能。相比之下 Mip-NeRF 投射锥体而不是射线，并对每个采样的锥台（此处显示为梯形）的体积进行明确建模，从而解决了这种歧义。

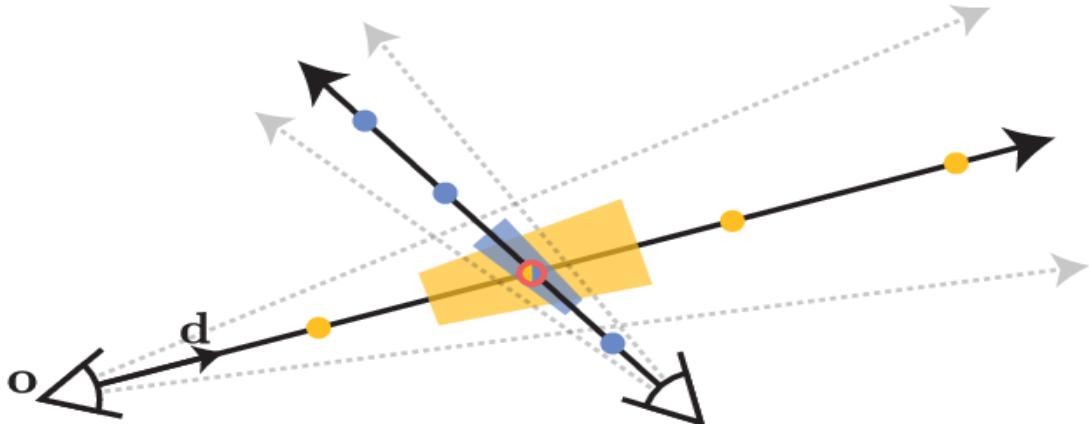


图 4: NeRF 采样方式缺陷

### 3.3 IPE 位置编码

mip-NeRF 的渲染和特征化过程中投射了一个圆锥体并沿着该圆锥体对圆锥台进行了特征化。与 NeRF 一样，mipNeRF 中的图像一次渲染一个像素，因此我们可以根据渲染的单个感兴趣像素来描述我们的过程。对于那个像素，我们从相机的投影中心  $\mathbf{o}$  沿着穿过像素中心的方向  $\mathbf{d}$  投射一个圆锥体。该圆锥体的顶点位于  $\mathbf{o}$ ，圆锥体在图像平面  $\mathbf{o} + \mathbf{d}$  处的半径参数化为  $\square r$ 。我们将  $\square r$  设置为按  $2/\sqrt{12}$  缩放的世界坐标中像素的宽度，这会产生一个圆锥体，其在图像平面上的截面在  $x$  和  $y$  方向上的方差与像素足迹的方差相匹配。位于两个  $t$  值  $[t_0, t_1]$  之间的圆锥台内的位置集（在图 1 中可视化）在一个范围中。理想情况下，这种特征化表示应该与 NeRF 中使用的位置编码特征具有相似的形式，表明这种特征表示对 NeRF 的成功至关重要 [30]。对此有许多可行的方法，但该文章发现的最简单和最有效的解决方案是简单地计算位于圆锥台内的所有坐标的预期位置编码，然而，尚不清楚如何有效地计算这样的特征，因为分子中的积分没有封闭形式的解。因此，用多元高斯近似圆锥台，这允许对所需特征进行有效近似，该编码方式称为“集成位置编码”（IPE）。

要用多元高斯近似圆锥台，我们必须计算  $F(x, \cdot)$  的均值和协方差。因为假定每个圆锥截头体都是圆形的，并且因为圆锥截头体围绕圆锥体的轴对称，所以这样的高斯具有三个（除了  $\mathbf{o}$  和  $\mathbf{d}$  之外）的完整特征：沿射线的平均距离，沿射线的方差，以及垂直于射线的方差。这些量根据中点  $t\mu = (t_0 + t_1)/2$  和半宽度  $t\delta = (t_1 - t_0)/2$  进行参数化，这对于数值稳定性至关重要。最终将这个高斯从圆锥台的坐标系转换为世界坐标，给我们最终的多元高斯。接下来，导出 IPE，它是根据上述高斯分布的位置编码坐标的期望值。生成 IPE 特征的最后一步是计算对这个提升的多元高斯分布的期望，由位置的正弦和余弦调制。这个预期的正弦或余弦只是被方差的高斯函数衰减的均值的正弦或余弦。有了这个，就可以将最终的 IPE 特征计算为协方差矩阵的均值和对角线的预期正弦和余弦。因为位置编码独立地对每个维度进行编码，所以这种预期编码仅依赖于  $\gamma(x)$  的边缘分布，并且只需要协方差矩阵（每个维度方差的向量）的对角线。如果直接计算这些对角线，IPE 特征的构建成本与 PE 特征大致相同。

图 5 可视化了玩具一维域中 IPE 和传统 PE 特征之间的差异。IPE 特征的行为很直观：如果位置编码中的特定频率的周期大于用于构造 IPE 特征的区间宽度，则该频率的编码不受影响。但是，如果周期小于间隔（在这种情况下，该间隔内的 PE 将反复振荡），则该频率的编码将按比例缩小至零。简而言之，IPE 保留了在一个区间内恒定的频率并温和地“移除”在一个区间内变化的频率，而 PE 保留了所有频率直到某个手动调整的超参数  $L$ 。通过以这种方式缩放每个正弦和余弦，IPE 功能有效地抗拒锯齿位置编码功能，可以平滑地编码空间体积的大小和形状。IPE 还有效地删除了  $L$  作为超参数，它可以简单地设置为一个非常大的值，然后永远不会调整。

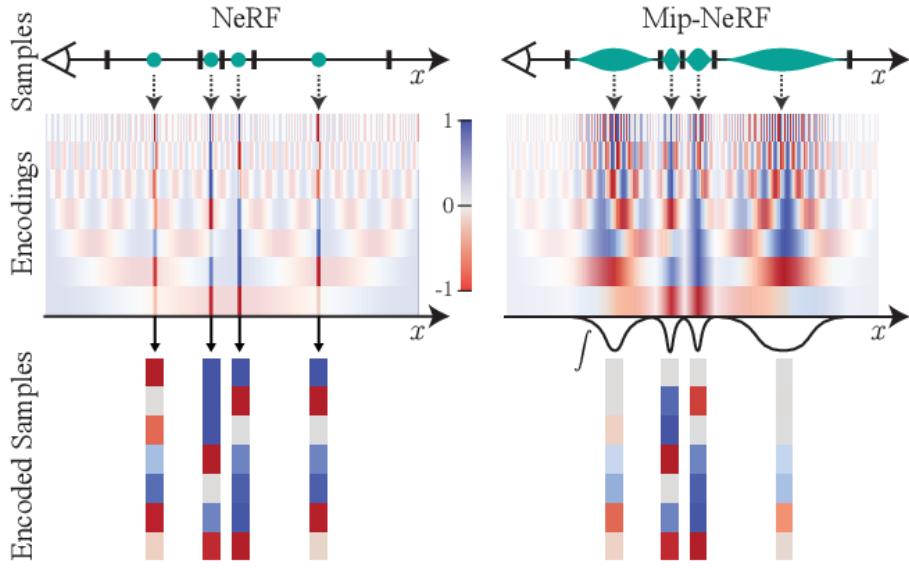


图 5: PE 和 IPE 的对比

## 4 复现细节

### 4.1 与已有开源代码对比

该论文复现是基于 NeRF-Pytorch 版本的官方源代码实现的，具体参考了源代码中数据处理，光线追踪以及 MLP 网络架构等，通过 Pytorch-lightning 来构建框架，来实现 NeRF 中的这些部分。其次，在论文方面，也参考了 Mip-NeRF 官方源码（JAX），通过研究其锥体采样部分的实现以及论文中对于 IPE 过程的公式来完成 Mip-NeRF 中 IPE 的部分。

### 4.2 实验环境搭建

创建 conda 环境，并进行配置，python 版本为 3.9。以及配置 cuda 安装所需要的库包：

`pytorch-lightning==1.5.2`

`einops==0.3.2`

`opencv-python==4.5.4.58`

`matplotlib==3.5.0`

`imageio==2.10.4`

`scipy==1.8.0`

`imageio-ffmpeg==0.4.5`

`tqdm==4.64.0`

`open3d==0.14.1`

运行 `train.py` 并配置相关参数即可。

### 4.3 创新点

本文的创新点在于：

1、将官方源码（JAX）版本转为 Pytorch 版本实现，并且尝试使用了 Pytorch-lightning 框架来规整代码，使得网络的实现以及代码的可读性都有一定程度的改进

2、修改了原先的激活函数，原始 NeRF 中，MLP 用于构造预测密度  $\tau$  和颜色  $c$  的激活函数分别是

ReLU 和 sigmoid。受到 Block-NeRF 的<sup>[2]</sup>启发代替 ReLU 作为产生  $\tau$  的激活函数，使用移位的 softplus:  $\log(1+\exp(x-1))$ 。softplus 中偏移-1 相当于将 mip-NeRF 中产生  $\tau$  的偏差初始化为-1，这会导致初始  $\tau$  值变小，这样会导致训练开始时的优化速度稍快一些。激活函数的这些变化对性能影响不大，但在使用大学习率时提高了训练稳定性。

3、损失函数，用 Smooth L1 损失代替原先的 L2 损失函数。L1 损失函数的鲁棒性要比 L2 强，L2 函数将真实标签与网络输出之间的误差进行了放大（二者之间误差大于 1 时）或缩小（二者之间误差小于 1 时），使用 L2 函数为训练目标的模型会对这种类型的样本更加敏感，在优化过程中模型不断调整适应这些样本，而这些样本可能本身是一个异常值，模型对这些异常值的优化适应则会导致其训练方向偏离目标。但是，L2 拥有比 L1 更光滑的曲线，更利于网络收敛，因为 L2 函数曲线光滑连续，处处可导，便于使用梯度下降法，这样也有利于收敛，能较快收敛到最小值。Smooth L1 则是在 L1 的基础上加入了 L2 处处可导的优点，使得在具有较强鲁棒性的情况下能和 L2 曲线一样处处可导，便于梯度下降。但是从实验效果上来看，对于实验效果的提升并不明显。

## 5 实验结果分析

在这次实验中，分别运行了 lego 模型的多尺度和单尺度模型。运行结果如图 6 所示。在实验结果中，分别可以看出用 Mip-NeRF 三维重建渲染后的深度图和体素网格，从而可以清楚的看到 Mip-NeRF 的实验结果以及抗锯齿效果还是比较理想的。下面还会用峰值信噪比和结构相似性<sup>[5]</sup>等数据来比较说明 Mip-NeRF 复现的效果。

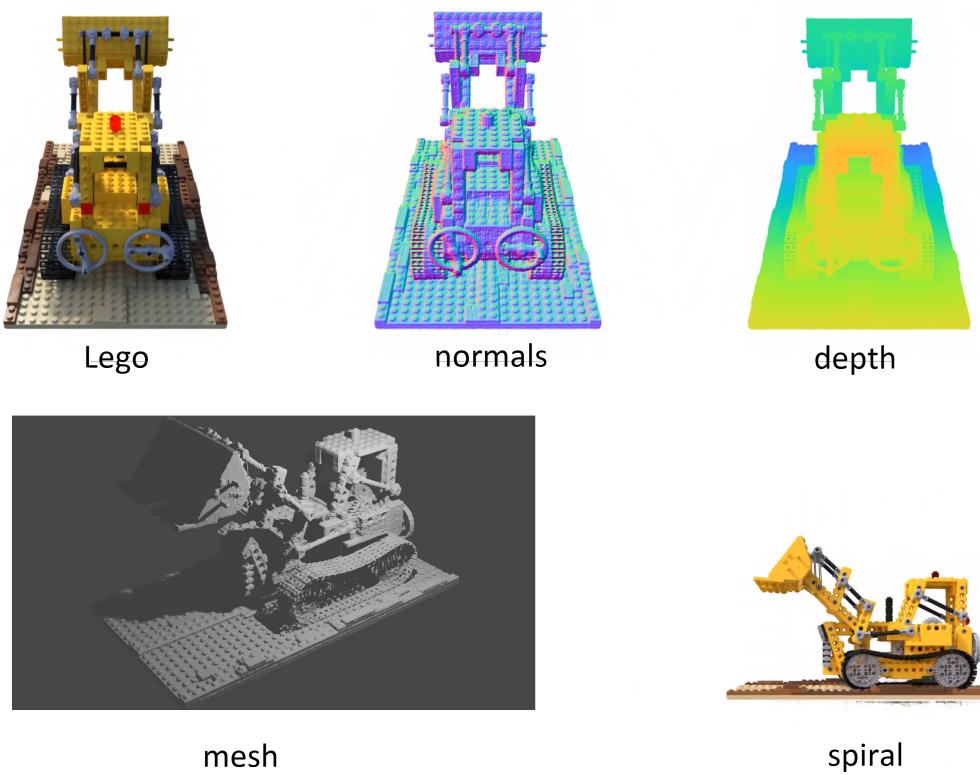


图 6: 实验结果

其中，多尺度模型的学习率以及峰值信噪比和训练时的 Loss 以及峰值信噪比如图 7 所示。从训练时的 Loss 可以看出，在创新中的激活函数还是起到了一定的作用，它使得一开始的优化速度稍快了

一些，虽然对最终结果帮助不大，但是对网络收敛也起到了一定的帮助，减少了一定的时间。其他的实验效果与 Mip-NeRF 官方源码（JAX）相比并没有太大差距，下面会用数据来说明复现版本与 JAX 版本的差距。

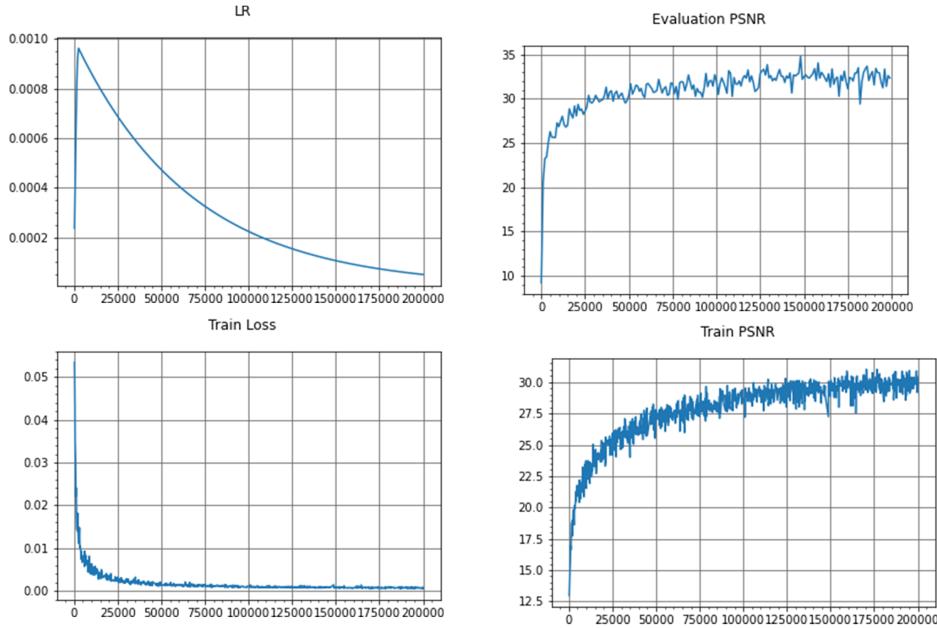


图 7: PE 和 IPE 的对比

从表 8 中可以看出，Pytorch 版本相较于 JAX 版本而言，在峰值信噪比方面有一点提升，在结构相似性方面则几乎一样。总体而言的改进并不大，但是实现起来还是相对容易一些。

	Multi Scale Train And Multi Scale Test										Single Scale		
	PNSR						SSIM					PNSR	SSIM
	Full Res	1/2 Res	1/4 Res	1/8 Res	Average (PyTorch)	Average (Jax)	Full Res	1/2 Res	1/4 Res	1/8 Res	Average (PyTorch)	Average (Jax)	Full Res
lego	35.231	36.439	36.664	35.578	35.978	35.736	0.975	0.986	0.990	0.991	0.986	0.9843	35.978 0.986

图 8: Pytorch 版本与 JAX 版本对比

最后是与 NeRF 官方源代码进行比较，如表 9 所示，和 Mip-NeRF 论文中展示出来的数据大致相同，并且在多分辨率中的差距也是比较明显。

	Multi Scale Train And Multi Scale Test									
	PNSR					SSIM				
	Full Res	1/2 Res	1/4 Res	1/8 Res	Average (PyTorch)	Full Res	1/2 Res	1/4 Res	1/8 Res	Average (PyTorch)
Mip	35.231	36.439	36.664	35.578	35.978	0.975	0.986	0.990	0.991	0.986
NeRF	27.471	28.016	27.816	26.657	27.491	0.918	0.931	0.936	0.931	0.929

图 9: NeRF 和 Mip-NeRF

## 6 总结与展望

本篇报告主要讲了对 Mip-NeRF 论文的复现过程，通过使用 pytorch 来实现对该论文的复现，并且用移位的 softplus 激活函数来代替原先的 sigmoid 激活函数，从而缩短模型收敛所需要的时间，并且

改进了损失函数，用 Smooth L1 代替原先的 L2 损失函数，尽管在最终实验结果上没有太大帮助，但是使得损失函数比原先的 L2 具有更强的鲁棒性。通过这次复现的结果和效果来看，复现的效果在峰值信噪比和相似性结构上都要略微优于原先的官方源码，由于还没有开始学习 JAX 框架，所以通过这次复现的机会，将 Mip-NeRF 用 Pytorch 来进行复现，也让我对于 Mip-NeRF 的流程以及其中的创新点和方法有了更加深入的理解。并且也对机器学习以及神经网络的过程有了更加深入的认识，包括网络结构和损失函数以及激活函数的使用。

后续希望自己可以通过学习更多的 NeRF 相关的知识，实现对 Mip-NeRF 的提速效果，虽然在这次的复现实现中已经有了相关的想法，但是没能很好的实现出来，还没有解决使用 Instant-nfp 或者是 mvsNeRF 来加速的适用问题，许多猜想都只在理论层面上觉得可行，但是实验起来却还是有许多问题，也让我对这些论文里的一些参数和变量以及他们所使用的方法有了比较深的理解。总之，前沿技术这门课程给了我一个很好的机会来更加深入的去了解我所学领域的相关技术，对论文的认识也更加清晰，让我受益匪浅。

## 参考文献

- [1] MILDENHALL B, SRINIVASAN P P, TANCIK M, et al. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis[C]//European Conference on Computer Vision. 2020.
- [2] TANCIK M, CASSER V, YAN X, et al. Block-NeRF: Scalable Large Scene Neural View Synthesis[J]. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022: 8238-8248.
- [3] BARRON J T, MILDENHALL B, TANCIK M, et al. Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields[J]. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021: 5835-5844.
- [4] AMANATIDES J. Ray tracing with cones[J]. Proceedings of the 11th annual conference on Computer graphics and interactive techniques, 1984.
- [5] WANG Z, BOVIK A C, SHEIKH H R, et al. Image quality assessment: from error visibility to structural similarity[J]. IEEE Transactions on Image Processing, 2004, 13: 600-612.