NeRF 论文的复现工作

张宇鹏

摘要

NeRF(Neural Radiance Field) 是最近提出的 3D 场景的隐式表达方法：MLP 做 3D 场景拟合，体渲染 (volume rendering) 生成逼真的新视角图像。由于细节的保真度好，NeRF 受到广泛的关注。本人的工作，是跑通 NeRF 的代码，复现实验效果，并尝试压缩训练时间：通过高斯分布拟合场景可视表面密度的分布，来减少 3D 点的采样数。

关键词：NeRF，高斯分布；

1 引言

现有的传媒模式，发布方全程制作，控制接收方的可见视角和范围。接收方无法自由选择感兴趣的观看视角。比如，世界杯比赛，用户希望至始至终从一个角度观看比赛，保持画面的连贯性。但是电视台讲解，为了讲解某些战术布局，强制切换画面视角。这样就打断了用户的连贯性。更进一步，也无法满足不同用户，有不同的观看角度和兴趣的需要。

人们感兴趣的视角，通常是多样的，可能分散在不同的角度。因此，在所有感兴趣的视角出布置相机，是不可行的。

那么，基于给定的角度的图像，合成新视角的图像，就体现出其价值。既能满足不同用户的不用视角的需求，也不增加设备的布置成本。Nerf[1] 要解决的就是新视角合成的问题：基于给定的图像，合成新视角的图像。

2 相关工作

最近，人们在尝试使用 mlp ，将一个空间位置，映射为一个形状的隐式表达，例如对于形状表面的有向距离。但是，这些方法在真实复杂场景的保真度方面，不如离散的表达方法效果好，例如三角网格或体素格子。本文提升神经场景表达的能力，来渲染逼真的复杂场景。

2.1 神经 3D 形状的表达

最近的工作，是用神经网络，将位置 xyz，映射到一个有向距离，或者占用场。此方法的缺点，是需要 3D 几何的真实数据。后续工作，推进到只需要 2D 图像来做训练。这些方法，现在只能达到低分辨率的简单几何的形状。本文提出的方法，可以优化网络，编码 5D 辐射场（3D 位置，2D 视角方向），从而表达一个更高分辨率的几何和表面，并且可以渲染出复杂场景的新视角的逼真图像。

2.2 视觉合成和基于图像的渲染

体积法 (Volumetric approaches) 可以表达逼真的复杂形状和材质，适合基于梯度下降的优化，和网格方法相比，可以产生更少的伪影。有的方法，结合 CNN (Convolution Neural Network) 和采样的体素格子，使得 CNN 补偿低分辨率的体素格子产生的离散伪影。这些方法合成效果好，但是无法应用到更高的分辨率，因为时间和空间复杂度很高。本文使用全连接神经网络，编码一个连续的物体。不仅在渲染质量上显著提升，在空间复杂度上降低明显。

2.3 训练时间的优化

NeRF 的训练时间长的十几个小时，甚至一两天，难以落地应用。已经有许多方法对 NeRF 进行优化[2]。本人基于人眼视觉的模式观察，尝试减少采样点，减少训练时间。

3 本文方法

3.1 本文方法概述

将一个连续场景，表达为向量值函数，输入是一个 5D (3D 是位置 X =(x, y, z)，2D 是视角方向 $d = (\theta, \phi)$)，输出是一个辐射出来的颜色 c = (r, g, b) 和体密度 σ 。使用一个 mlp 网络 F_Θ 逼近一个连续的 5D 场景表达。其中 $F_\Theta : (X, d) \rightarrow (c, \sigma)$ 。通过优化参数 Θ ，来达到映射关系。

由于密度是物体本身的性质，无视角无关，因此只需要位置 X ，就可以输出形状的密度 σ 。而颜色 c ，和位置有关，也和视角有关，因此依赖于 (X, d) 。神经网络结构如图 1 所示：

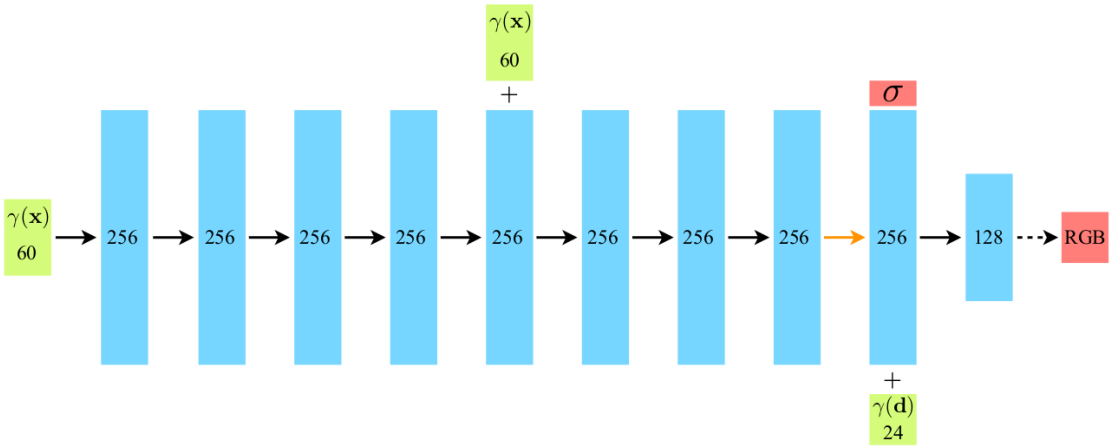

图 1: 神经网络结构图

3.2 辐射场的体渲染

5D 神经辐射场，将一个场景，表达为一个体密度和有向辐射场。使用经典的体渲染 (volume rendering) 方法，渲染穿过场景的每一条光线的颜色。体密度 $\sigma(x)$ 可以解释为，光线在位置 x 处因为无限小的粒子终止的概率。相机光线 $r(t) = o + td$ 的颜色 $C(r)$ ，可以表示为

$$C(r) = \int_{t_n}^{t_f} T(t)\sigma(r(t))c(r(t), d)dt, \text{ 其中} T(t) = exp(-\int_{t_n}^{t} \sigma(r(s))ds)$$

t_f, t_n 表示光线的考虑范围的边界。$\sigma(r(t))$ 表示在位置 t 处的体密度，由神经网络输出。$c(r(t), d)$ 表示在位置 t 处辐射出来的颜色，由神经网络输出。

3.3 体渲染的分层采样优化

位置 t ，是采样的点。如果太稀疏，则难以拟合出真实的场景。如果太密集，则计算量会非常庞大。分层采样，可以解决此问题。在一条射线上，先做一次均匀的粗采样（64 个采样位置），通过粗神经网络得到场景的体密度的大概分布。再根据体密度的大概分布，进行细致的位置采样（128 个采样位置），汇总粗-细采样位置，在细神经网络做查询。有限的采样位置，更逼近真实的体密度分布。

3.4 尝试的改进方法

人眼观察场景时，只能看到物体最表面的一层。每个视角下，我们只关心最近的表面。因此，通过一个高斯分布，拟合一个视角下的一条光线上的最靠近视角点的体密度的分布。根据拟合的分布，

来采样位置，可以用更少的点，表达视角下可见的表面的信息。从而查询更少次数的神经网络，减少训练时间。

4 复现细节

4.1 复现工作

github 有 pytorch 版本的复现代码：https://github.com/yenchenlin/nerf-pytorch。在一张 P100 上，训练一个场景的 400 * 400 分辨率的图像，需要 200k 迭代，约 12 小时。

4.2 尝试改进的实现

每个图像的每个像素 i 的光线上的可见的体密度分布，使用高斯分布 $N(\mu_i, \sigma_i)$ 近似。采样区间是 $[2, 6]$，所有像素初始化为 $N(4, 5)$，则在采样区间上，近似均匀分布。通过神经网络拟合一次后，得到一条光线上每个 3D 点对渲染颜色的贡献 $w = T(t)\sigma((r(t))$，根据贡献占比，估计光线上的新的分布参数 μ_i', σ_i'，如此迭代更新。本人新增的代码，主要在 nerf-pytorch/run_nerf.py 文件中，使用 zyp 做了代码块的标记。

5 实验结果分析

复现代码，可以得到场景的新视角的合成图像，如图 2。渲染出来的效果录制成视频，记录迭代次数为 (50k，100k，150k，200k) 时的渲染效果，见视频文件《13-张宇鹏指导老师-周杰.mkv》。实验证明，改进的方法，不能保证训练收敛，因此，并没有达到减少训练时间的效果。

由于 nerf 的 mlp，是表达整个场景的 3D 信息，不仅仅是投影后的 2D 信息，因此只考虑视角下的可见体密度，不能和 nerf 的 mlp 匹配。导致无法收敛。

图 2: 渲染效果图。左边，10k 迭代，0.5h；中间，100k 迭代, 6h；右边，200k 迭代，12h

6 总结与展望

第一次对前沿的论文的方法尝试改进。观察人眼的视觉模式，尝试改进 nerf 的训练时间。但是实验证明，思路是不通的。以后，再参考其他的训练时间方面的优化，借鉴学习。

参考文献

[1] MILDENHALL B, SRINIVASAN P P, TANCIK M, et al. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis[J]., 2020.

[2] GAO K, GAO Y, HE H, et al. RNeRF: Neural Radiance Field in 3D Vision, A Comprehensive Review [J]. arXiv:2210.00379, 2022.