NeRF in the Wild论文阅读笔记

1. 引言

论文地址：

https://arxiv.org/pdf/2008.02268v3.pdfarxiv.org/pdf/2008.02268v3.pdf

NeRF在视图合成方面取得的成功有目共睹，然而NeRF表现优秀的需要在一些特定的前提下：

1. 图像的拍摄间隔不大，以保证不同图像之间的光强保持一致；
2. 图像中的场景是静态的，没有运动与遮挡。

当这两个前提遭到破坏，NeRF的表现会有显著的下降，这也阻碍了NeRF在户外场景下的直接应用。因此NeRF要求同一个位置和朝向的图像是独一无二的，这和很多真实世界的数据集相矛盾。不同的摄影师可能在同一地点拍摄不同的图像，照片中会有汽车、人等移动的物体，也会有季节更替，昼夜交替。即便是同一时刻拍摄的图像也会有曝光、颜色、色调等由于相机不同和图像处理带来的变化。

作者提出了NeRF-W模型，将图像中的外观变化部分如曝光、光线、天气等建模到低维的空间中，并使用**Generative Latent Optimization**框架优化出一系列外观嵌入组件(appearance embedding)，这样在应对变化的图像时就能具有很强的灵活性。将场景中的元素分为两类：**共享元素**和**图像相关元素**，并且采用**无监督**的方法将场景元素分为**静态**和**瞬态**两类。于是在渲染过程中能够去除瞬态的元素进而合成更为真实准确的场景图像。

2. 相关工作

- Novel View Synthesis
- Neural Rendering

3. 背景知识

具体背景知识可以先阅读理解NeRF：

https://arxiv.org/pdf/2003.08934.pdfarxiv.org/pdf/2003.08934.pdf

4. NeRF in the Wild

类似于 NeRF，作者实现的 NeRF-W 从没有限制的图片集$\{I_i\}_{i=1}^N$中学习一个体密度表示F_θ。NeRF 有两个很重要的假设，不同图片之间的光强一致性以及静态的场景。而对于网络上的图片，往往都无法满足这样的条件。

- **光度变化**：在户外摄影，一天中所处的时间以及当时的气象条件都会影响图像光度的变化。此外，相机中的曝光、白平衡、色调匹配等设置都会加剧图像光度的不一致。
- **瞬态对象**：真实世界中很少捕获到孤立的地标，往往会伴随着本身的移动或者其他物体的遮挡。特别是旅游胜地的照片拍摄尤为困难，因为总是会伴随着行人的移动与遮挡。

于是作者提出了两种模型组件来分别解决这两种问题。

4.1 潜在表面建模

通过 Generative Latent Optimization (GLO) 框架学习图像内部的空间结构潜在变量，由图像I_i生成表面编码向量$l_i^{(a)}$，提供了对每张图像外观变化的一种解释性表示，这使得模型可以灵活地适应不同图像的光照和外观变化。。在 NeRF 中推导$c(t)$的网络如下：

$$c(t) = \mathrm{MLP}_{\theta_2}(z(t), \gamma_d(d)) \tag{4}$$

其中 (z(t)) 为经过 8 层网络隐藏层后与σ同时产生的特征信息。而 NeRF-W 中改进后的网络为：

$$c_i(t) = \mathrm{MLP}_{\theta_2}(z(t), \gamma_d(d), l_i^{(a)}) \tag{7}$$

这样的设计不会影响到另一个网络的σ，即不会影响到场景的 3D 几何形状。如果将$l_i^{(a)}$的长度$n^{(a)}$限制得小一些，那么模型倾向于优化出一个连续的空间，能够嵌入不同的光照条件，并且外观向量表示的特征更具概括性，模型在不同图像中能找到较一致的光照、曝光、白平衡等外观特性。这种泛化能力能够使模型在不同环境光照、相机参数下仍能生成一致的 3D 场景。；如果 $n^{(a)}$太大，外观嵌入向量可能会记住每张图像的特定细节，而不是学习到能广泛泛化的外观特征。这会导致模型在训练数据上表现良好，但在未见过的数据上表现较差，即过拟合。通过限制$n^{(a)}$的大小，可以强迫模型只学习到对图像外观变化的最重要的特征，而不是每个细节。

其中左1和右1都是训练图片，包含行人信息。

4.2 瞬态物体

在NeRF-W中可以添加两个组件来解决瞬态物体的遮挡。首先，原来MLP模型中用以生成颜色信息的网络层称为“静态”头，作者在模型中加入一个“瞬态”头用以估计瞬态物体的颜色信息。此外，允许“瞬态”头对每个像素的颜色信息保持怀疑，产生一个类似于颜色和密度一样的场来表示不确定性，使用**极大似然估计**来预测像素上颜色值。这两部分组件可以让模型无监督地区分静态和瞬态物体。

于是根据新的网络重新写颜色渲染方程：

$$\hat{C}_i(r) = \sum_{k=1}^{K} T_i(t_k) \left(\alpha(\sigma(t_k)\delta_k) c_i(t_k) + \alpha(\sigma_i^{(\tau)}(t_k)\delta_k) c_i^{(\tau)}(t_k) \right) \tag{8}$$

$$T_i(t_k) = \exp\left(- \sum_{k'=1}^{k-1} (\sigma(t_{k'}) + \sigma_i^{(\tau)}(t_{k'}))\delta_{k'} \right) \tag{9}$$

将静态和瞬态部分相加并且均添加一个α系数。

将观察到的颜色值$C_i(r)$拟合为 **球形高斯分布** (isotropic normal distribution)，方差为$\beta_i(r)^2$以及均值$\hat{C}_i(r)$。这里的方差也通过类似渲染的方式计算得到：

$$\hat{\beta}_i(r) = \mathcal{R}(r, \beta_i, \sigma_i^{(\tau)}) \tag{10}$$

下面给出瞬态部分网络的表达式：

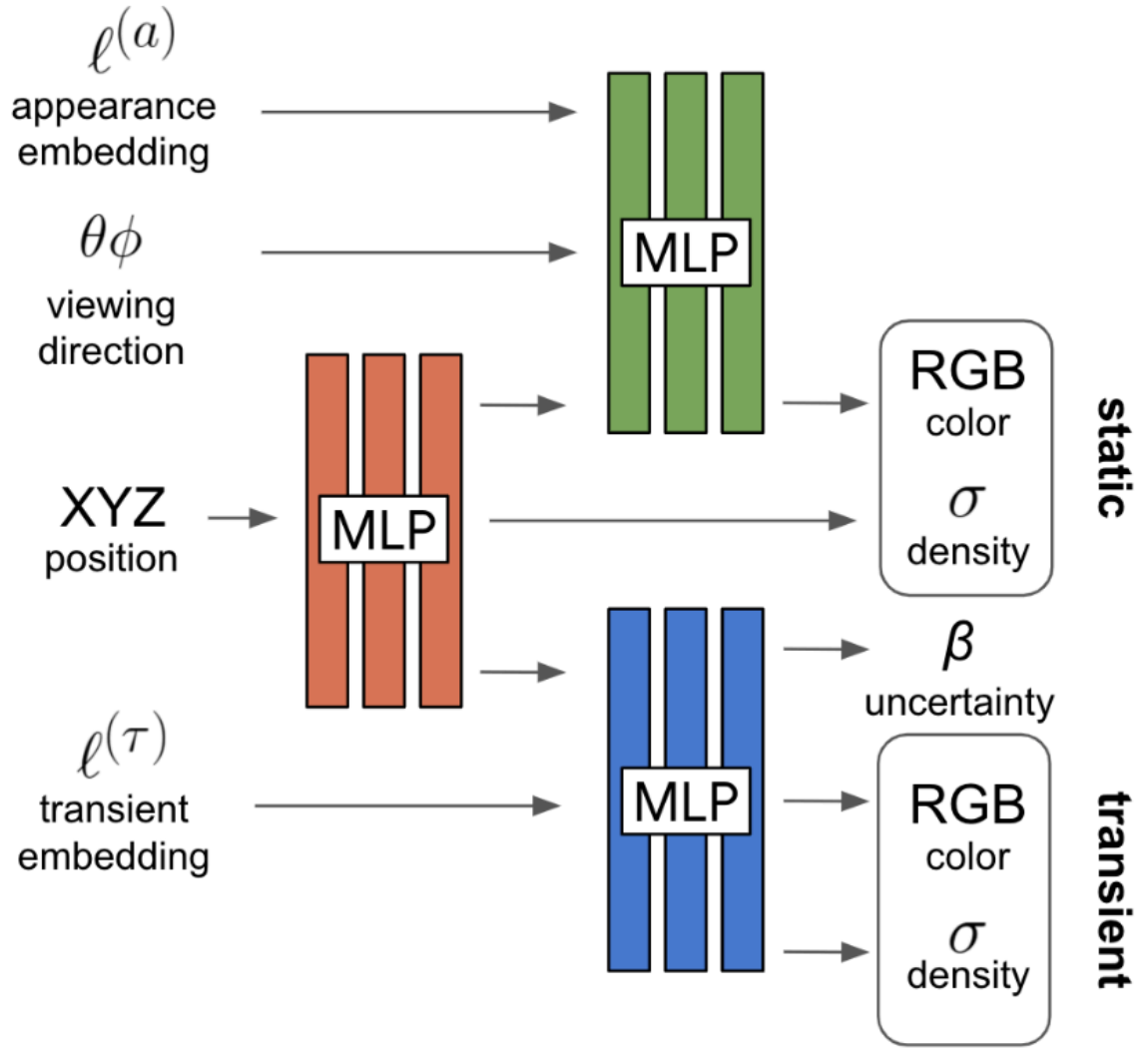
$$[\sigma_i^{(\tau)}(t), c_i^{(\tau)}(t), \tilde{\beta}_i(t)] = \mathrm{MLP}_{\theta_3}(z(t), l_i^{(\tau)}) \tag{11}$$

$$\beta_i(t) = \beta_{\min} + \log\left(1 + \exp\left(\tilde{\beta}_i(t)\right)\right) \tag{12}$$

如果不设置最小值，模型可能会将$\beta_i(r)$优化得过小，从而让损失函数的第一项（平方误差项）有更大的影响力。这样，模型会倾向于精确地拟合某些特定像素的预测值，甚至可能会过度拟合噪声或瞬时现象，导致模型在训练数据上表现很好，但在测试数据上泛化能力差。通过给$\beta_i(r)$设置最小值，可以防止模型过度相信那些有噪声或难以预测的像素，减少过拟合的风险。并并且可以保证不会完全忽略任何一条射线。

设置最小值能够确保模型在所有像素上至少有一个合理的不确定性估计。即使对于那些模型认为非常可靠的预测，也需要有一个最低的不确定性水平。现实中的数据通常带有一定的噪声和不确定性，完全消除不确定性是不现实的。因此，设置一个最小值确保模型不会将不确定性降到不合理的低水平。

对于$\sigma_i^{(\tau)}(t)$通过一层ReLU函数激活,$c_i^{(\tau)}(t)$通过sigmoid函数激活，而$\tilde{\beta}_i(t)$通过softplus函数激活。



上图即为改进过后的网络架构图，绿色为推理静态物体的头，蓝色为新增推理瞬态物体的头，都额外增加了 **embedding** 作为输入。

作者为该模型定义了新的损失函数：

$$L_i(r) = \frac{\|C_i(r) - \hat{C}_i(r)\|_2^2}{2\beta_i(r)^2} + \frac{\log \beta_i(r)^2}{2} + \lambda_u \frac{1}{K} \sum_{k=1}^{K} \sigma_i^{(\tau)}(t_k) \tag{13}$$

为了使模型的预测$\hat{C}_i(r)$尽可能逼近真实值$C_i(r)$，我们可以通过最大化该高斯分布的似然来调整模型参数。负对数似然形式为：

$$L = -\log P(C_i(r)|\hat{C}_i(r), \beta_i(r)^2)$$

对于高斯分布的概率密度函数，负对数似然的具体形式为：

$$L = \frac{\|C_i(r) - \hat{C}_i(r)\|_2^2}{2\beta_i(r)^2} + \log \beta_i(r)$$

- 第一项是 **负对数似然**：将预测颜色$\hat{C}_i(r)$和真实颜色$C_i(r)$之间的差异建模为一个高斯分布。这里，方差为$\beta_i(r)^2$表示该像素的不确定性。如果$\beta_i(r)$大，意味着模型认为该像素不可靠。
- 第二项是 **正则化项**：与高斯分布的对数分区函数有关，防止$\beta_i(r)$变得过大。这确保模型不会简单地将不确定性无限增加来降低损失。
- 第三项是 **瞬时密度的 L1 正则化**：这项用于防止模型依赖瞬时物体的密度来解释静态场景中的现象。通过惩罚非零的瞬时密度$\sigma_i^{(\tau)}(t)$，模型被鼓励将大部分信息归入静态部分，而不是瞬时部分。这意味着模型会倾向于解释场景中的大部分区域为静态部分，而瞬时部分只在真正有变化的地方才会被赋予非零值。

该损失函数的第一部分是根据希望拟合的概率分布的均值和方差构造的，但是倾向于学习到一个很大的$\beta_i(r)$，大的方差降低了该像素对应颜色$C_i(r)$的可能性，于是用第二部分来权衡。第三部分是一个 L1 正则化，这会倾向于让模型不用瞬态的密度$\sigma_i^{(\tau)}(t)$去表征静态的场景。

5. 实验结果

	Brandenburg Gate			Sacre Coeur			Trevi Fountain			Taj Mahal			Prague			Hagia Sophia		
	PSNR	MS-SSIM	LPIPS	PSNR	MS-SSIM	LPIPS	PSNR	MS-SSIM	LPIPS	PSNR	MS-SSIM	LPIPS	PSNR	MS-SSIM	LPIPS	PSNR	MS-SSIM	LPIPS
NRW [22]	23.85	0.914	0.141	19.39	0.797	0.229	20.56	0.811	0.242	21.24	0.844	**0.201**	19.89	0.803	**0.216**	20.75	0.796	**0.231**
NeRF	21.05	0.895	0.208	17.12	0.781	0.278	17.46	0.778	0.334	15.77	0.697	0.427	15.67	0.747	0.362	16.04	0.749	0.338
NeRF-A	27.96	0.941	0.145	24.43	0.923	0.174	26.24	0.924	0.211	25.99	0.893	0.225	22.52	0.870	0.244	21.83	0.820	0.276
NeRF-U	19.49	0.921	0.174	15.99	0.826	0.223	15.03	0.795	0.277	10.23	0.778	0.373	15.03	0.787	0.315	13.74	0.706	0.376
NeRF-W	**29.08**	**0.962**	**0.110**	**25.34**	**0.939**	**0.151**	**26.58**	**0.934**	**0.189**	**26.36**	**0.904**	0.207	**22.81**	**0.879**	0.227	**22.23**	**0.849**	0.250

Table 1: Quantitative results on the Phototourism dataset [13] for NRW [22], NeRF [24], and two ablations of the proposed model. Best results are **highlighted**. NeRF-W outperforms the previous state of the art across all datasets on PSNR and MS-SSIM and achieves competitive results in LPIPS. Note that LPIPS generally favours methods such as NRW trained with an adversarial or perceptual loss and it is less sensitive to typical GAN artifacts, see Figures 7 and 14 (supplementary).

由上图可以明显看到NeRF-W的效果相对于先前版本的NeRF有了较大的提升。

Figure 7: Qualitative results from experiments on the Phototourism dataset. NeRF-W is simultaneously able to model appearance variation (top), remove transient occluders (flag, middle), and reconstruct fine details in the scene (bottom). Further datasets are shown in Figure 14 (supplementary). Photos by Flickr users firewave, clintonjeff, leoglenn_g / CC BY.

这里可以看到NeRF-W能够很好地还原细节，具备较高的分辨率，同时能够生成与ground truth相近的图像明度。