Zip-NeRF 个人笔记

前言

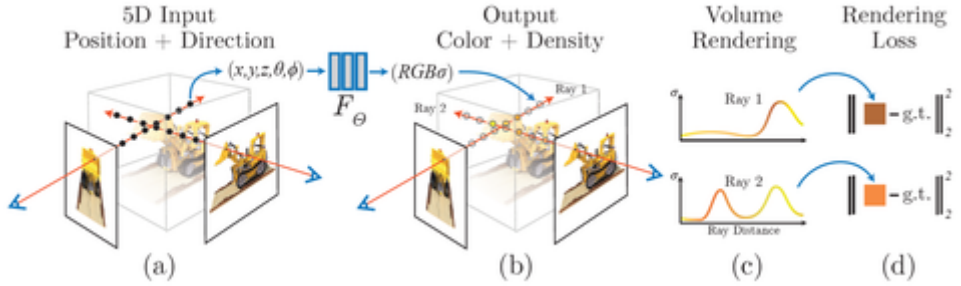
Barron 大佬最新力作，可以看作 Mip-NeRF 360++。

Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields

https://jonbarron.info/zipnerf/jonbarron.info/zipnerf/

看 Zip-NeRF 之前得对 Mip-NeRF、Mip-NeRF 360 以及 Instant-NGP 有一定的了解。可以看一下我之前的笔记中相关部分粗略地了解一下。

ỉ 景页：NeRF系列工作个人总结928 赞同 · 73 评论文章



> 这里推荐理解如何从频率的角度出发看 NeRF 的采样和 Encoding，这对理解这篇文章很有帮助。个人觉得 Instant-NGP 这些个 Hybrid 方法的网格存储 (Hash ENcoding) 的值可以视作是某种基函数在该点的离散采样。高分辨率等同于高采样率，这也就意味着高分辨率的网格可以存取高频基函数的离散点采样结果，相对的低分辨率的网格则存取低频基函数的离散点采样结果。简单来说就是，细网格存高频，粗网格存低频信息。如果能够接受这一点，就能够非常丝滑地从 Mip-NeRF 360 的角度出发接受 Zip-NeRF。

Zip-NeRF 结合了 Mip-NeRF 360 和 iNGP（谷歌居然接受 Instant-NGP 了，打不过就加入？），而它做的事情也十分简单，简单来说就是将 Mip-NeRF 360 中关于截断视锥/截头体 (frustum) 的 Fourier Encoding 的期望换成关于截断视锥的 Hash Encoding 的期望，同时改进了一下 Proposal MLP。而 Zip-NeRF 也从 Anti-Aliasing 出发来讲好这个故事，将这两个贡献点点串了起来。

整篇文章非常工程（所以感觉一作是 Barron 有些不可思议），涉及到的数学公式也非常的简单（相比于 Mip-NeRF 和 Mip-NeRF 360 而言）。 这篇文章，我们最重要理解它要做的事情以及思路，具体的细节非常的繁琐（里面的表述和数学公式），可以后面再回过头来看。

Zip-NeRF 的贡献由两部分组成：Sampling & Encoding 和 Proposal MLP Supervision，对应下面两章节。而 Zip-NeRF 提到的 Anti-Aliasing 则是文章包装故事的重要支撑点，我们先暂时放到一篇，在下面有需要时再提及。下面的解读依旧从如何求得关于截断视锥的 Hash Encoding 的期望开始说起。

Spatial Anti-Aliasing

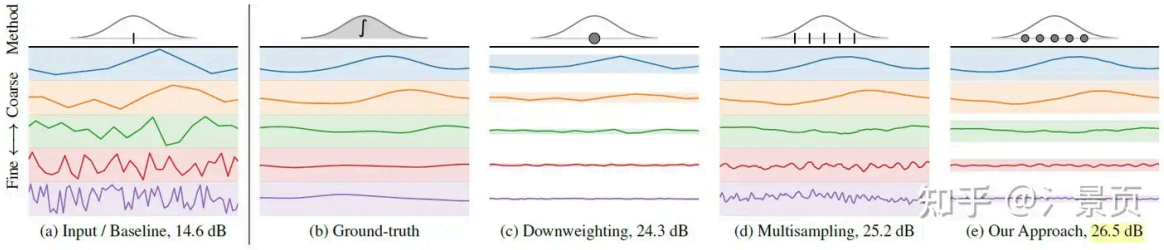
这一部分讲实现关于截断视锥的 Hash Encoding 的期望的计算，实现了这一目标就具备了 Prefilter 的能力，从而实现了 Spatial Anti-Aliasing。在 Mip-NeRF 中提到，原始 NeRF 没有 Prefilter 的能力，会在高频编码部分产生失真（近大远小 — 随着视线的延申，远处的截断视锥不应该有显著的高频编码分量，但 NeRF 做不到），而 Mip-NeRF 利用高斯近似计算出截断视锥的编码期望作为截断视锥的编码信息，具备了 Prefilter 的能力，从而克服了原始 NeRF 中的高频编码分量的失真现象。

在 Zip-NeRF 中同理，作者认为原始的 iNGP 也存在类似的失真现象（尤其对于细网格存储的特征而言），因此作者如法炮制，希望将 Mip-NeRF 中的那一套搬到 iNGP 中来克服这一失真现象（只是在实现过程中，有些许的区别）。

作者首先介绍了一下 iNGP 的失真情况和对应的解决方案及其结果，其实就是回顾了一下 Mip-NeRF 的内容（根据 iNGP 的插值过程出了一套可视化结果）并做了一些拓展。

- 图 (a) 其实就是 iNGP（不要被他的 Gaussian's mean 唬住了，就是采一个样本点），可以看到其非常地"生硬"，此时，分段的线段就对应着索引网格特征的线性插值操作，线段两端的顶点就对应着网格存储的编码信息。注意高频部分（紫色）存在非常令人不悦的毛刺，而这个现象实际上就对应着 Mip-NeRF 中提到的原始 NeRF 的高频分量失真现象；
- (b) 所示的是理想情况，经过完美的高斯滤波得到非常平滑且合理的特征（不同频率信号的强度由高斯滤波的带宽控制，高斯滤波带宽越大，高频信号衰减程度越大），但在实际过程中实现完美的高斯滤波显然不太现实；
- (d) 表示粗暴的多点采样策略（Multisampling 其实就是 Supersampling），即在高斯滤波的带宽范围内（这样描述其实不对，理解一下大概意思就好）采多个点，然后将采样点的信号的均值作为均值点位置的信号，可以看到整个信号确实变得平滑了，但是高频部分的毛刺依旧存在；
- (c) 则是 Mip-NeRF 单点高斯期望和 iNGP 的简单结合，可以看到 Mip-NeRF 的策略确实起到了 Prefilter 的效果（信号强度随着频率的升高衰减），但由于 Hash Encoding 的线性插值过程，其信号依旧呈现分段的形式。

综上，Zip-NeRF 希望得到平滑的信号，同时又能根据高斯滤波的带宽来控制高频信号的幅度（Prefilter 能力），因此，最直接的方法就是将 Mip-NeRF (c) 和 Multisampling (d) 缝合起来，通过计算多个样本点的高斯期望的均值作为均值点位置的信号。而 Zip-NeRF 就是这么做的，如下图 (e) 所示，这一策略确实得到了非常不错的结果（可以说是非常工程了）。



> Figure: Here we show a toy 1-dimensional iNGP [23] with 1 feature per scale. Each subplot represents a different strategy for querying the NGP at all coordinates along the x axis — imagine a Gaussian moving left to right, where each line is the NGP feature for each coordinate, and where each color is a different scale in the iNGP. (a) The naive solution of querying the Gaussian's mean results in features with piecewise-linear kinks, where the high frequencies past the bandwidth of the Gaussian are large and inaccurate. (b) The true solution, obtained by convolving the NGP features with a Gaussian — an intractable solution in practice — results in coarse features that are smooth but informative and fine features that are near 0. (c) We can suppress unreliable high frequencies by downweighing them based on the scale of the Gaussian (color bands behind each feature indicate the downweighting), but this results in unnaturally sharp discontinuities in coarse features. (d) Alternatively, supersampling produces reasonable coarse scales features but erratic fine-scale features. (e) We therefore multisample isotropic sub-Gaussians (5 shown here) and use each sub-Gaussian's scale to downweight frequencies.

以上可以看出 Zip-NeRF 在 Encoding 部分的想法非常简单和直白 — **采截断视锥的多个点当作高斯，计算它们的编码信号期望，然后平均得到截断视锥的编码信号期望**，这也是图 (e) 表示的意思。

值得注意的是，求取期望的对象的改变 (Fourier Encoding -> Hash Encoding) 使得截断视锥从各项同性变成了各向异性（Hash Encoding 对空间进行了不同细粒度的网格**划分**，不再具有 Fourier Encoding 那种连续平滑且周期的性质，这在上图中 (a/c) 的分段现象也有所体现），所以不能够再简单地套用用 Mip-NeRF 的 basis lifting 策略进行期望近似计算。为此，Zip-NeRF 利用混合高斯模型，通过多个各向

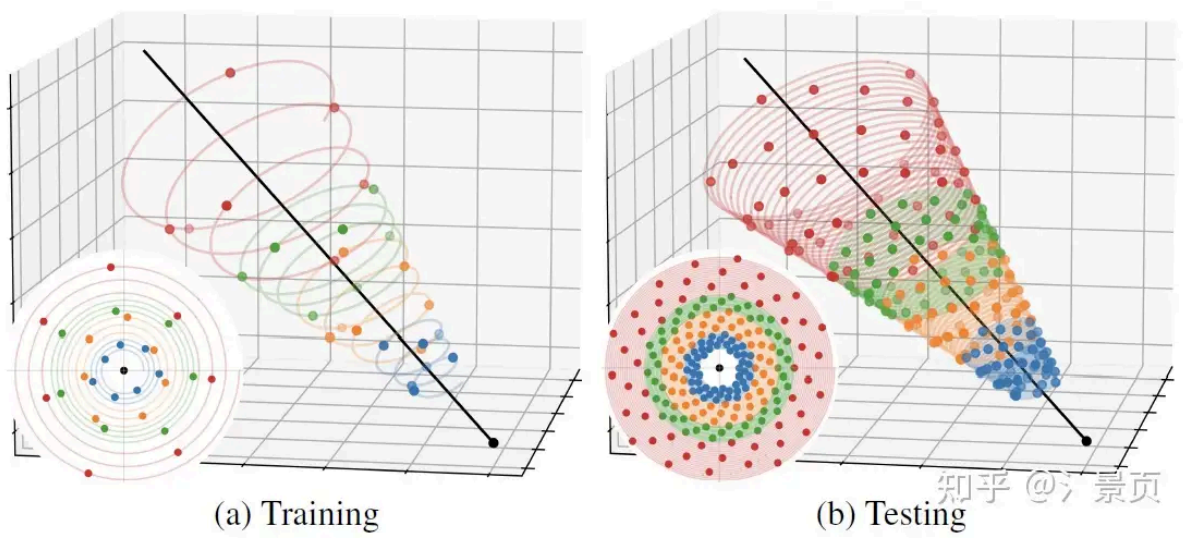
同性的高斯来近似这个各向异性的截断视锥，具体的操作则是如上图 (e) 所示。这里得清楚文章中的 sub-volume 对应的就是截断视锥（所以为什么不好好统一叫 frustum。。。）。

接下来的这张图比较令人迷惑，为什么是在缠绕视锥的螺旋线上采点呢？为什么不在内部采点呢？这里得提到文中一个非常重要的假设：**Hash Encoding 中网格的特征期望为 0。**而这里写得非常令人难受，在文中第二章第二段毫无征兆地抛出 *This isotropic assumption lets us approximate the true integral of the feature grid over a sub-volume by leveraging the fact that the values in the grid are zero-mean.* 后，在后面的 Downweighting 小节才提到 Zip-NeRF 约束了网格学习的特征期望尽可能为 0 从而满足这一假设。

总之，在满足了这一假设后，我们在**计算截断视锥的编码信息期望时就可以完全无视被截断视锥包含的网格的编码信息，只需关心那些被截断视锥部分包含的网格的编码信息即可，而这些被部分包含的网格，恰恰就是围绕着视锥分布的。**为了近似这些网格产生的编码信息，Zip-NeRF 选择视锥表面上采点。简单点来说就是内部点采了等于白采，所以只采边缘的点。

那为什么是在缠绕视锥的螺旋线上采点呢？Zip-NeRF 特意提到了每个截断视锥区域的螺旋线的圈数 mmm 和样本点数 nmnn 得是互质的 (co-prime)。个人理解是，足够的圈数保证了样本点产生的各向同性高斯能够更好地近似 z 轴方向上的各向异性（此时的 z 轴指视角方向），互质保证了样本点产生的各向同性高斯能够更好地近似 xy 方向上的各向异性（不互质感觉点数够了影响也不大？）。

> 个人感觉主要还是样本点够了就行。



> Figure: Here we show a toy 3D ray with an exaggerated pixel width (viewed from afar and viewed along the ray as an inset) divided into 4 frustums denoted by color. We multisample each frustum by placing nmnn points along a spiral with mmmm loops such that the sample mean and covariance of multisamples exactly matches the frustum's true mean and covariance. (a) During training we use a small randomized set of multisamples for each frustum, (b) but at test time we use a large deterministic set of multisamples.

通过上面的分析可知，我们在获得了样本点后需要将样本点转换成高斯再计算其编码期望。值得注意得是，将样本点转换成高斯再计算其编码期望的过程本质上是希望实现 Prefilter，而这个 Prefilter 又表现为近大远小的能力，简单来说就是越远编码特征衰减越大，那最简单的方式就是给一个随着视线延申变小的权值就好了。

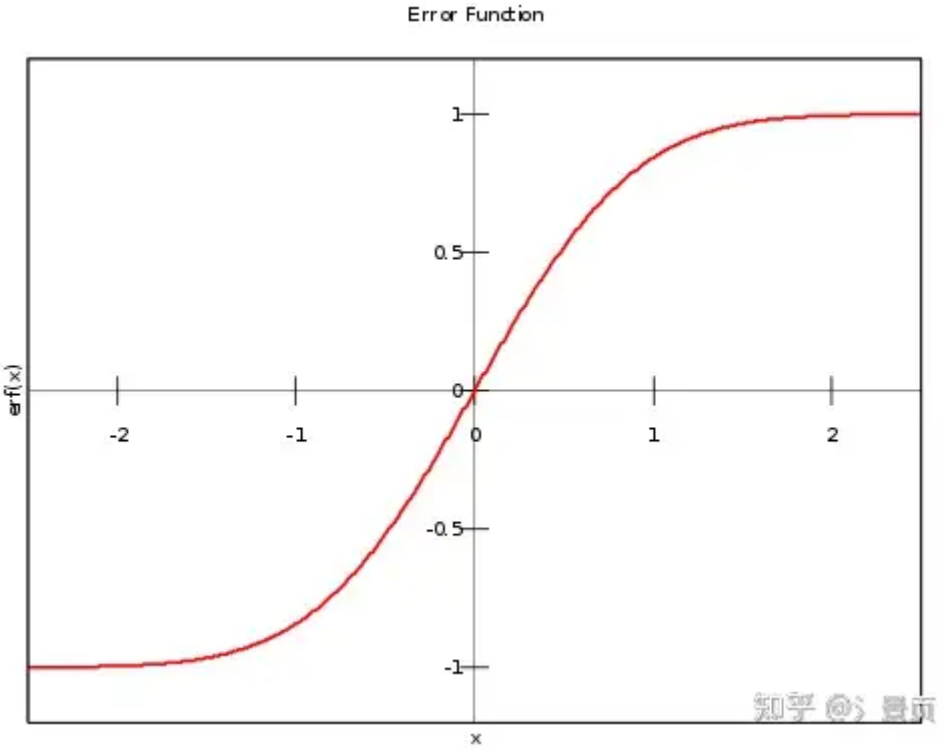
这段内容提到了 Zip-NeRF 模型中的权值计算方法。具体来说，作者通过三线性插值获取样本点的编码特征，并为编码信息的每个 level 级别设置一个权值 (\omega_{j, l})，这个权值是样本点在视线延申中缩小的趋势的反映。公式中，(j) 表示视线中的样本点，(l) 表示 Hash 编码中的网格级别。

权值的计算公式为：

$$\omega_{j,l} = \text{erf}\left(\frac{1}{\sqrt{8\sigma_j^2 n_l^2}}\right)$$

其中，erf 是误差函数（Error Function），定义为：

$$\text{erf}(x) = \frac{1}{\sqrt{\pi}}\int_{-x}^{x} e^{-t^2}dt = \frac{2}{\sqrt{\pi}}\int_{0}^{x} e^{-t^2}dt$$



其中样本点的标准差 $\sigma_j = rt$ 会随着视角方向的远离而变大（其实就是把这里的样本点看成了 Mip-NeRF 中的高斯球）；网格的边长 n_l 会随着 level 的提升而变小。

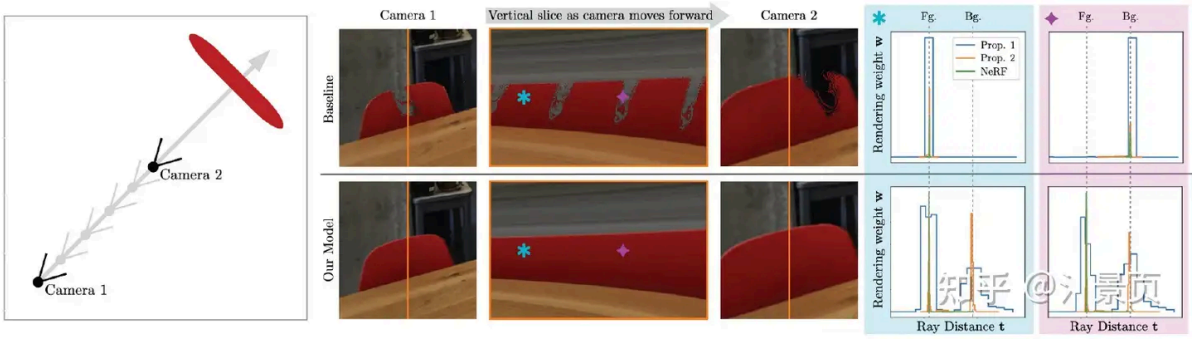
以上就可知道 $\omega_{j,l}$ 在同一 level 的网格n_l固定）中，会随着视线方向的远离而衰减，实现了 prefilter 的功能。此外当 σ_j固定时，网格$level$越高，即n_l 越小时，$\omega_{j,l}$也越大，个人理解为作者还是比较希望高频部分的信息能够尽可能多地参与训练。以上，截断视链的编码期望就可以通过对这些采样点的编码期望均值得到：

$$f_l = \text{mean}_j(\omega_{j,l} \cdot \text{trilerp}(n_l \cdot \mathbf{x}_j; V_l))$$

Z-Aliasing and Proposal Supervision

这一部分由于介绍得太过繁琐，还没有看完，在这里先说一下他想解决的事情和思路。具体的细节可能会之后再进行补充。

Zip-NeRF 在这里指出了 Mip-NeRF 360 中 Proposal MLP 产生的关于 z轴（视线方向）的失真问题，如下图所示。

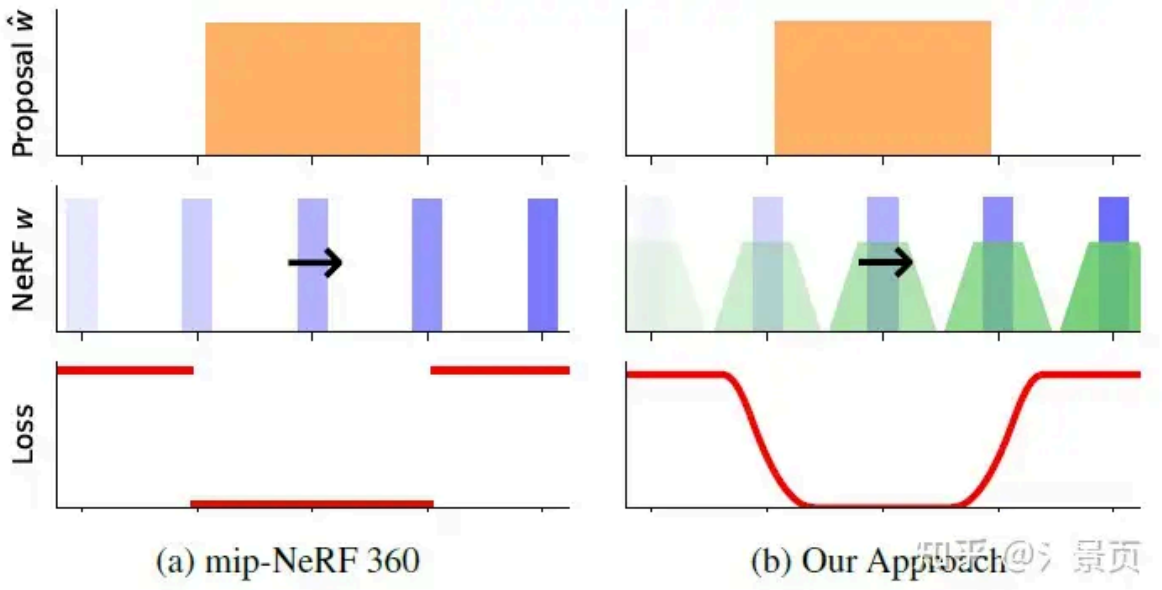


> Figure: Here we visualize the problem of z-aliasing. Left: we have a scene where 2 training cameras face a narrow red chair in front of a gray wall. Middle: As we sweep a test camera between those training cameras, we see that the baseline algorithm (top) “misses” or “hits” the chair depending on its distance and therefore introduces tearing artifacts, while our model (bottom) consistently “hits” the chair to produce artifact-free renderings. Left: This is because the baseline (top) has learned non-smooth proposal distributions due to aliasing in its supervision, while our model (bottom) correctly predicts proposal distributions that capture both the foreground and the background at all depths due to our anti-aliased loss function.

这里强烈建议大家去看 Zip-NeRF Project Page 的 Z aliasing 视频，视频动态地展示了拉近相机时产生的失真效果，非常直观。

造成这种失真的原因则是由于 Mip-NeRF 360 中对 Proposal MLP 的监督是由 NeRF MLP 预测转换得到的，而这个监督信号如下图所示是分段不连续的，无法监督到 Proposal MLP 采样间隔的内部（文章中称之为 interlevel loss）。

> 至于为什么不连续会产生这样的失真，我下面会给出我自己的直观解释。



> Figure: Here we visualize proposal supervision for a toy setting where a narrow NeRF histogram (blue) translates along a ray relative to a coarse proposal histogram (orange). (a) The loss used by mip-NeRF 360 is piecewise constant, but (b) our loss is smooth because we blur NeRF histograms into piecewise linear splines (green). Our prefiltered loss lets us learn anti-aliased proposal distributions.

为此，Zip-NeRF 改进了 Mip-NeRF 360 中的 Proposal MLP，选择在监督的时候利用样条函数进行平滑处理。

这部分讲得很繁琐，我给出我自己的算是比较直观的频率角度解释。简单点说就是 Proposal MLP 的采样间隔具有明显的宽度（如图 4 中蓝色和橙色），在训练视角不够密集的情况下，对于 Proposal MLP 来说（尤其是第一个 Proposal MLP，即蓝色部分），这段宽度范围内是无法分辨的（见的次数不多，采样宽度又较大，可以理解为采样率不够）。所以当相机沿着 z 轴就在这段宽度范围内移动，Proposal MLP 预测的深度可能会发生剧烈的变化。而 Zip-NeRF Project Page 的 Z aliasing 视频中左边可以明显看到 Mip-NeRF 360 的这个失真呈现一种非常显著的周期性，这个周期刚好就是第一个 Proposal MLP 的采样间隔。

为了更方便理解，我将采样间隔中的小区域称之为“次区间”（类似次像素的概念），而 Zip-NeRF 的改进则是通过滤波或者插值的方式生成这个“次区间”的监督。这样就能够使 Proposal MLP 产生平滑的预测结果，极大地改善失真现象。

而从 Zip-NeRF Project Page 的 Z aliasing 视频中右边也可以看到，虽然这个 Proposal MLP 的密度分布还是会随着视角的拉近呈现一种周期性的变动，但显然这个变动已经变得非常地平滑了。

> 但粗暴来说，如果训练视角给的够密，结合 MLP 的插值性质，这一失真应该就不会出现。