Paper Summary Mip-NeRF

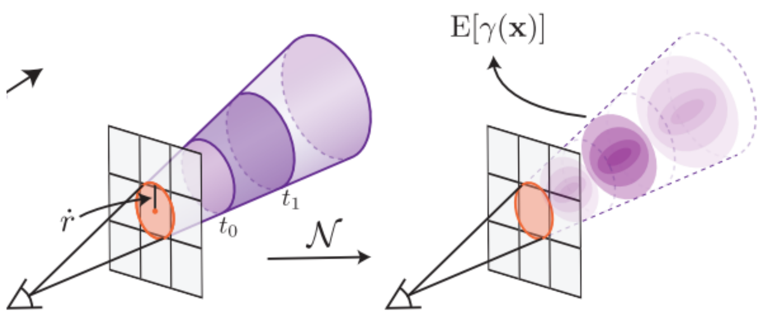
3. Method

A pixel's color is the integration of all incoming radiance within the pixel's frustum. NeRF casts a single infinitesimally narrow ray per pixel, resulting in aliasing. Mip-NeRF ameliorates this issue by casting a **cone** from each pixel.

Dividing the cone being cast into a series of **conical frustums**, and using **integrated** positional encoding (IPE)

3.1 Cone Tracing and Positional Encoding

Describing procedure in terms of an individual pixel of interest being rendered. Casting a cone from the camera's center of projection $\mathbf{o}$ along the direction $\mathbf{d}$ that passes through the pixel's center. The **radius** of the cone at the image plane $\mathbf{o}+\mathbf{d}$ is parameterized as $\dot{r}$



b) Mip-NeRF

The set of positions $\mathbf{x}$ that lie within a conical frustum between two t values $[t_0, t_1]$ is:

$$F(\mathbf{x}, \mathbf{o}, \mathbf{d}, \dot{r}, t_0, t_1) = 1\left\{\left(t_0 < \frac{\mathbf{d}^{\mathbf{T}}(\mathbf{x}-\mathbf{o})}{||\mathbf{d}||_2^2} < t_1\right) \wedge \left(\frac{\mathbf{d}^T(\mathbf{x}-\mathbf{o})}{||\mathbf{d}||_2||\mathbf{x}-\mathbf{o}||_2} > \frac{1}{\sqrt{1+(\dot{r}/||\mathbf{d}||_2)^2}}\right)\right\}$$

Prove is in the inference file.

Then we must construct a featurized representation of the volume inside this conical frustum. Simply compute the expected positional encoding of all coordinates that lie within the conical frustum:

$$\gamma^*(\mathbf{o}, \mathbf{d}, \dot{r}, t_0, t_1) = \frac{\int \gamma(\mathbf{x}) F(\mathbf{x}, \mathbf{o}, \mathbf{d}, \dot{r}, t_0, t_1) dx}{\int F(\mathbf{x}, \mathbf{o}, \mathbf{d}, \dot{r}, t_0, t_1) dx}$$

It's difficult on computation, therefore, approximate the **conical frustum** with a **multivariate Gaussian**, called **IPE**.

This Gaussian is fully characterized by three values (in addition to $\mathbf{o}$ and $\mathbf{d}$): the mean distance along the ray μ_t, the variance along the ray σ_t^2, and the variance perpendicular to the ray σ_r^2

......

We can transfer this Gaussian from the coordinate frame of the conical frustum into world coordinates as follows:

$$\mu = \mathbf{o} + \mu_t \mathbf{d}$$

$$\Sigma = \sigma_t^2(\mathbf{d}\mathbf{d}^T) + \sigma_r^2 \left(\mathbf{I} - \frac{\mathbf{d}\mathbf{d}^T}{||\mathbf{d}||_2^2}\right)$$

give us final multivariate Gaussian, so here this is how we approximate the conical frustum.

Next, we derive the IPE, which is the expectation of a positionally-encoded coordinate distributed according to the previous Gaussian. Firstly, rewrite the PE in Equation 1 as Fourier Feature

$$\begin{bmatrix} 1 & 0 & 0 & 2 & 0 & 0 & & 2^{L-1} & 0 & 0 \\ 0 & 1 & 0 & 0 & 2 & 0 & \dots & 0 & 2^{L-1} & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & & 0 & 0 & 2^{L-1} \end{bmatrix}$$

$$\gamma(x) = \begin{bmatrix} sin(\mathbf{P}x) \\ cos(\mathbf{P}x) \end{bmatrix}$$

This reparameterization allows us to derive a closed form for IPE. Using the fact $Cov[Ax, By] = ACov[x,y]B^T$, we can identify the mean and covariance of our conical frustum Gaussian

$$\mu_\gamma = \mathbf{P}\mu$$

$$\Sigma_\gamma = \mathbf{P}\Sigma\mathbf{P}^T$$

Final Step: computing the expectations over this lifted multivariate Gaussian, modulated by the sine and cosine of position. These expectations have simple closed-form expressions:

$$E_{x\sim N(\mu,\sigma^2)}[sin(x)] = sin(\mu)exp(-(\frac{1}{2})\sigma^2)$$

$$E_{x\sim N(\mu,\sigma^2)}[cos(x)] = cos(\mu)exp(-(\frac{1}{2})\sigma^2)$$

With this we can compute our final IPE features as the expected sines and cosines of the mean and the diagonal of the covariance matrix:

$$\begin{aligned} \gamma(\mu, \Sigma) &= E_{x\sim N(\mu_\gamma, \Sigma_\gamma)}[\gamma(x)] \\ &= \begin{bmatrix} sin(\mu_\gamma) \circ exp(-(\frac{1}{2})\ diag(\Sigma_\gamma)) \\ cos(\mu_\gamma) \circ exp(-(\frac{1}{2})\ diag(\Sigma_\gamma)) \end{bmatrix} \end{aligned}$$