A. Color estimation (MLP head)

Let view direction $v \in \mathbb{R}^3$ (unit), spatial position x, SH embedding $\Phi_{\mathrm{SH}}(v)$ up to degree D, hash-grid position encoding $\psi(x)$, and per-Gaussian material vector m.

$$f_{\mathrm{in}}(x,v) = \left[\, \Phi_{\mathrm{SH}}(v) \,\|\, \psi(x) \,\|\, m \,\right], \tag{1}$$
$$\tilde{c} = \mathrm{MLP}\big(f_{\mathrm{in}}\big), \tag{2}$$
$$c = \mathrm{sigmoid}(\tilde{c}). \tag{3}$$

B. Transmittance-driven visibility (replaces α-sorting)

B1. Clip ray to unit cube $\mathcal{B} = [0,1]^3$

Given ray $r(t) = p + t\,v$, $t \in [0,1]$, for each axis $j \in \{x,y,z\}$ and planes $x_j \in \{0,1\}$:

$$t_j^{(0)} = \frac{0 - p_j}{v_j}, \quad t_j^{(1)} = \frac{1 - p_j}{v_j}, \quad t_j^{\min} = \min\{t_j^{(0)}, t_j^{(1)}\}, \quad t_j^{\max} = \max\{t_j^{(0)}, t_j^{(1)}\}. \tag{4}$$

Aggregate entrance/exit and clamp:

$$t_{\mathrm{enter}} = \max_j t_j^{\min}, \qquad t_{\mathrm{exit}} = \min_j t_j^{\max}. \tag{5}$$

$$t_0 = \mathrm{clip}(t_{\mathrm{enter}}, 0, 1), \quad t_1 = \mathrm{clip}(t_{\mathrm{exit}}, 0, 1), \quad \lambda = \max\{t_1 - t_0, 0\}. \tag{6}$$

If $\lambda > 0$, the clipped segment is

$$s = p + t_0 v, \qquad w = \lambda v, \qquad L = \|w\|_2. \tag{7}$$

B2. Midpoint sampling and segment transmittance

Choose step parameter $\sigma > 0$; set

$$K = \left\lceil \frac{L}{\sigma} \right\rceil, \qquad \Delta s = \frac{L}{K}, \qquad s_{\mathrm{factor}} = \frac{\Delta s}{\sigma}. \tag{8}$$

Midpoints and features:

$$\tilde{t}_k = \frac{k + \frac{1}{2}}{K}, \qquad x_k = s + \tilde{t}_k\, w, \qquad z_k = \big[\phi(x_k),\, u\big], \tag{9}$$

where ϕ is a hash feature and u collects any extra attributes. A scalar network outputs

$$a_k = \mathrm{net}(z_k). \tag{10}$$

Per-step factor and segment transmittance:

$$\rho_k = \big(\mathrm{sigmoid}(a_k)\big)^{s_{\mathrm{factor}}}, \qquad T = \prod_{k=0}^{K-1} \rho_k. \tag{11}$$

B3. Compositing

For Gaussian i with base opacity $\mathrm{opacity}_i$ and color c_i,

$$\alpha_i = \mathrm{opacity}_i \cdot T_i, \qquad C = \sum_i \alpha_i \, c_i, \qquad S = \sum_i \alpha_i. \tag{12}$$

with background b,

$$I = \begin{cases} \dfrac{C}{S}, & S > 1, \\ C + (1 - S)\, b, & S \le 1. \end{cases} \tag{13}$$

C. Multi-image-tile rendering (batched training)

Tiling. Partition the image plane into square tiles $\{\mathcal{B}_t\}_{t=1}^{K}$ of side S, with integer origin $o_t \in \mathbb{Z}^2$:

$$\mathcal{B}_t = \{\, p \in \mathbb{R}^2 \mid o_t \le p < o_t + (S, S) \,\}. \tag{14}$$

Tile-relative shift. For a Gaussian with screen center x_i and support radius $r_i > 0$,

$$x_i^{(t)} = x_i - o_t, \qquad \mathcal{P}_t = \{\, i \mid \mathrm{dist}(x_i, \mathcal{B}_t) \le r_i \,\}. \tag{15}$$

Batched launch. Use a 3D CUDA grid over per-tile pixels on (x, y) and tiles along z. The batched output tensor is

$$Y \in \mathbb{R}^{K \times C \times S \times S} \tag{16}$$

stored contiguously by tile; base write offset for tile t is $t \cdot (CSS)$.