Rendering

Jinyuan Wu

August 12, 2023

1 What a renderer does

A completely accurate surface renderer does the following:

- It takes the following input:
  - The positions and shapes of boundaries in the system;
  - BSDF or BSSDF of each boundaries;
  - Positions and directional distributions of light sources.
- It aims to calculate the radiance $L(\boldsymbol{r}, \omega)$ for given $\boldsymbol{r}$ and ω;[1] at boundaries that allow transmission we need to distinguish between $L(\boldsymbol{r}, \omega)$ on the two sides of the boundary; we can use L_i and L_o to refer to them.

  Since the radiance is conserved along a ray (Pharr et al. chap. 14), we can also use $L(\boldsymbol{r} \to \boldsymbol{r}')$ to refer to the radiance, if we are able to find some $\boldsymbol{r}$ and $\boldsymbol{r}'$ on boundaries,[2] and $\widehat{\boldsymbol{r}' - \boldsymbol{r}} = \omega$. With this notation, the i and o subscripts are no longer necessary.
- It enumerates all possible light paths that ends with a specific point $\boldsymbol{r}$. For each path,
  - If it comes directly from a light source $\boldsymbol{r}'$, then it contributes $L_\mathrm{e}(\boldsymbol{r}' \to \boldsymbol{r})$ to $L(\boldsymbol{r}, \omega)$, where L_e is the emission radiance of the light source.
  - If reflection or transmission happens at $\boldsymbol{r}'$, the contribution of $\boldsymbol{r}'' \to \boldsymbol{r}' \to \boldsymbol{r}$ is

    $$\begin{aligned} \mathrm{d}L\left(\boldsymbol{r}' \to \boldsymbol{r}\right) &= f(\boldsymbol{r}', \widehat{\boldsymbol{r}' - \boldsymbol{r}''}, \widehat{\boldsymbol{r} - \boldsymbol{r}'}) \lvert\cos\theta\rvert \, L(\boldsymbol{r}'' \to \boldsymbol{r}') \, \mathrm{d}\omega'' \\ &= f(\boldsymbol{r}', \widehat{\boldsymbol{r}' - \boldsymbol{r}''}, \widehat{\boldsymbol{r} - \boldsymbol{r}'}) \lvert\cos\theta\rvert \, L(\boldsymbol{r}'' \to \boldsymbol{r}') \frac{\mathrm{d}A''}{\lvert \boldsymbol{r}'' - \boldsymbol{r}' \rvert^2}, \end{aligned} \tag{2}$$

    where

    $$\cos\theta = \hat{\boldsymbol{n}}' \cdot \widehat{\boldsymbol{r}' - \boldsymbol{r}''}, \tag{3}$$

    and $\hat{\boldsymbol{n}}'$ is the normal vector at $\boldsymbol{r}'$.

    Of course, now the problem is how to find $L(\boldsymbol{r}'' \to \boldsymbol{r}')$; if $\boldsymbol{r}''$ is on a light source, (2) is already evaluated, and if not, we need to evaluate $L(\boldsymbol{r}'' \to \boldsymbol{r}')$ on the LHS of (2).
- $L(\boldsymbol{r}, \omega)$ is then used together with a camera model to create a "photon" of the system.

Since we need to sum over all rays determined by geometrical optics, this is known as ray tracing.

2 Camera model

[1] Here I use the notation in computer graphics: ω is the direction of the wave vector, and $\mathrm{d}\omega$ is the solid angle element surrounding the direction ω; we define

$$\delta(\omega - \omega') = \frac{1}{\sin\theta} \delta(\theta - \theta') \delta(\varphi - \varphi'). \tag{1}$$

[2] Often we need to deal with lights from infinity, but at least in theory we can model "infinity" by a large light source at a distance.