Radiosity Engine

Lighting Engine

Let $\vec{\epsilon}$ be the emissions vector (ϵ_i = emission of face F_i) for the current color, F be the list of faces, $n = |F|$, ρ be the reflectivity vector (ρ_i = reflectivity of face F_i) for the current color, R be the scene radius, and V be the visibility matrix ($V_{ij} = 1$ if faces F_i and F_j can be connected by a straight line which doesn't pass through another face, and $V_{ij} = 0$ otherwise).

Recall that the solution function u to the rendering equation for the current color can be approximated by:

$$\tilde{u_j} A_j = A_j \epsilon_j + \sum_{i=1}^{n} \tilde{u_i} B_{ji} \implies (1 - \sum_{i=1}^{n} \frac{B_{ij}}{A_j}) \tilde{u_j} = \epsilon_j \implies (I - C)\tilde{u} = \epsilon$$

Where we've defined C_{ij} to be $\frac{B_{ij}}{A_j}$. Then the entries of the matrix C can be determined from:

$$\begin{aligned} C_{ij} &= \rho_j \frac{A_j}{9} \int_{\vec{z} \in F_j} \int_{\vec{x} \in F_i} \frac{(\hat{n}_{F_i} \cdot (\vec{z} - \vec{x}))(\hat{n}_{F_j} \cdot (\vec{x} - \vec{z}))}{|\vec{x} - \vec{z}|^4} \, V_{ij} \, d\vec{x} \, d\vec{z} \\ &\approx V_{ij} \rho_j \frac{A_j}{9} \sum_{k=1}^{3} \sum_{l=1}^{3} S_{ijkl} \end{aligned}$$

$$\text{where } S_{ijkl} = \begin{cases} 0 & |\vec{F}_{ik} - \vec{F}_{jl}| < 0.005 \cdot R \\ \frac{(\hat{n}_{F_i} \cdot (\vec{F}_{ik} - \vec{F}_{jl}))(\hat{n}_{F_j} \cdot (\vec{F}_{jl} - \vec{F}_{ik}))}{|\vec{F}_{ik} - \vec{F}_{jl}|^4} & |\vec{F}_{ik} - \vec{F}_{jl}| \geq 0.005 \cdot R \end{cases}$$

Where $\vec{n}_{F_i}$ is the outwards pointing normal vector from face F_i, and $\vec{F}_{ik}$ is the kth vertex (expressed as a vector) of the face F_i (faces are triangles, so $1 \leq k, l \leq 3$). The outwards pointing normal vector can be computed by normalizing $\vec{n}_{F_i} = (\vec{F}_{i2} - \vec{F}_{i1}) \times (\vec{F}_{i3} - \vec{F}_{i1})$ since the vertices are listed counterclockwise.

From here it is possible to directly compute C given F, ϵ, ρ, V, and R, then solve $(I - C)\tilde{u} = \epsilon$ using a matrix equation solving software. The implementation differs slightly due to how the list of faces was formatted, but it follows the same guidelines for computing $\tilde{u}_r, \tilde{u}_g, \tilde{u}_b$, before scaling them to fit in the range $(0, 1)$ and returning them.