02564 Real-Time Graphics

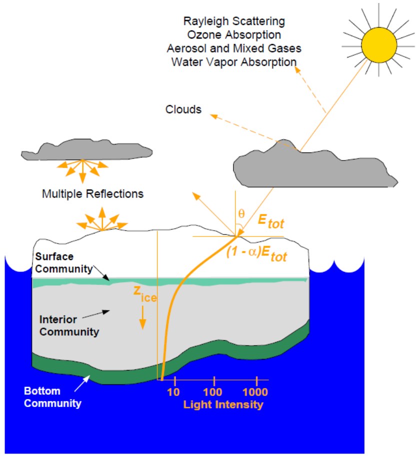
Skylight and irradiance environment maps

Jeppe Revall Frisvad

March 2016

Skylight to enhance outdoor scenes

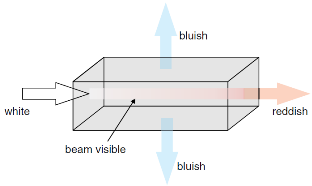
The atmosphere

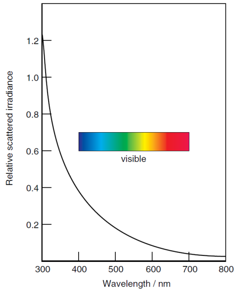


Reference

- Belém, A. L. Modeling Physical and Biological Processes in Antarctic Sea Ice. PhD Thesis, Fachbereich Biologie/Chemie der Universität Bremen, February 2002.

Rayleigh scattering





- Quote from Lord Rayleigh [On the light from the sky, its polarization and colour. *Philosophical Magazine 41*, pp. 107–120, 274–279, 1871]:

  If I represent the intensity of the primary light after traversing a thickness x of the turbid medium, we have

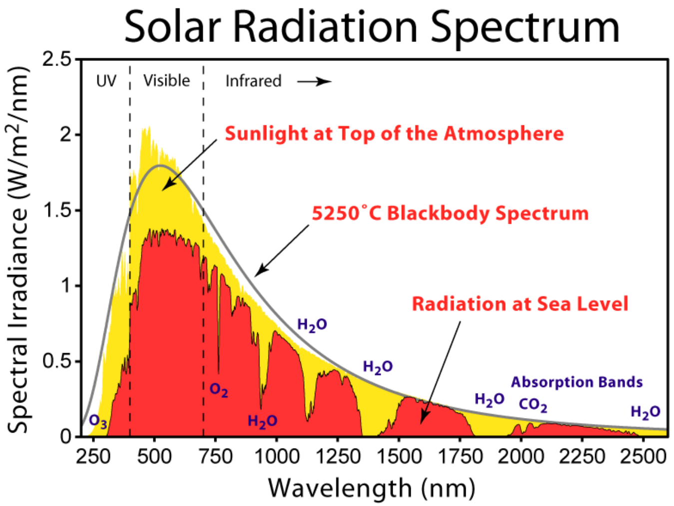
  $$dI = -kI\lambda^{-4}\,dx\ ,$$

  where k is a constant independent of λ. On integration,

  $$I = I_0 e^{-k\lambda^{-4}x}\ ,$$

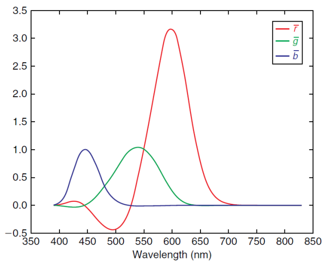
  if I_0 correspond to $x = 0$, —a law altogether similar to that of absorption, and showing how the light tends to become yellow and finally red as the thickness of the medium increases.

Solar radiation

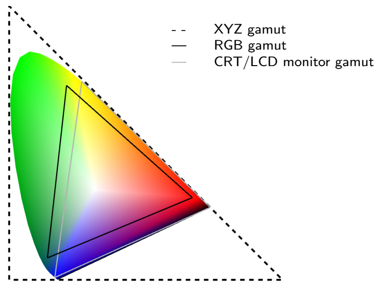


[Source: http://en.wikipedia.org/wiki/Sunlight]

Colorimetry



CIE color matching functions



The chromaticity diagram

$$
\begin{aligned}
R &= \int_{\mathscr{V}} C(\lambda)\bar{r}(\lambda)\,\mathrm{d}\lambda \\
G &= \int_{\mathscr{V}} C(\lambda)\bar{g}(\lambda)\,\mathrm{d}\lambda \\
B &= \int_{\mathscr{V}} C(\lambda)\bar{b}(\lambda)\,\mathrm{d}\lambda\ ,
\end{aligned}
$$

where $\mathscr{V}$ is the interval of visible wavelengths and $C(\lambda)$ is the spectrum that we want to transform to RGB.

Gamut mapping

- Gamut mapping is mapping one tristimulus color space to another.
- Gamut mapping is a linear transformation. Example:

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} .
$$

$$
\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.2405 & -1.5371 & -0.4985 \\ -0.9693 & 1.8760 & 0.0416 \\ 0.0556 & -0.2040 & 1.0572 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}
$$

- Y in the XYZ color space is called *luminance*.
- Luminance is a measure of how bright a scene appears.
- From the linear transformation above, we have

$$
Y = 0.2126\,R + 0.7152\,G + 0.0722\,B\,.
$$

Dynamic range

- Ambient luminance levels for some common lighting environments:

Condition	**Illumination (cd/m^2)**
Starlight	10^{-3}
Moonlight	10^{-1}
Indoor lighting	10^{2}
Sunlight	10^{5}
Maximum intensity of common monitors	10^{2}

Reference

- Reinhard, E., Ward, G., Pattanaik, S., Debevec, P., Heidrich, W., and Myszkowski, K. High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting, second edition, Morgan Kaufmann/Elsevier, 2010.

Tone mapping

- Simplistic tone mapping: scale and gamma correct:

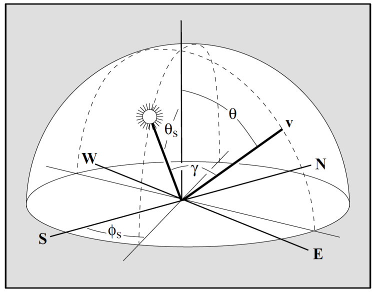
$$(R', G', B') = \left((sR)^{1/\gamma}, (sG)^{1/\gamma}, (sB)^{1/\gamma}\right) ,$$

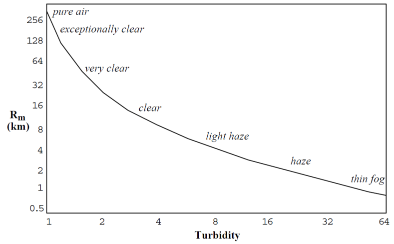
  where s and γ are user-defined parameters.
- The framework uses $s = 0.025$ (and $\gamma = 1$) for the sun and sky.
- Another tone mapping operator (Ferschin's exponential mapping):

$$(R', G', B') = \left((1 - e^{-R})^{1/\gamma}, (1 - e^{-G})^{1/\gamma}, (1 - e^{-B})^{1/\gamma}\right) .$$

- This is useful for avoiding overexposed pixels.
- Other tone mapping operators use sigmoid functions based on the luminance levels in the scene [Reinhard et al. 2010].

Analytical sky models [Preetham et al. 1999] (input parameters)





- Solar declination angle:

$$\delta = 0.4093 \sin\left(\frac{2\pi(J - 81)}{368}\right) .$$

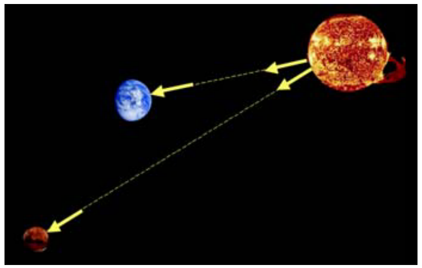
- Solar position:

$$\theta_s = \frac{\pi}{2} - \arcsin\left(\sin\ell \sin\delta - \cos\ell\cos\delta\cos\frac{\pi t}{12}\right) ,$$

$$\phi_s = \arctan\left(\frac{-\cos\delta\sin\frac{\pi t}{12}}{\cos\ell\sin\delta - \sin\ell\cos\delta\cos\frac{\pi t}{12}}\right) ,$$

  where $J \in [1, 365]$ is Julian day, t is solar time, and ℓ is latitude.

Direct sunlight



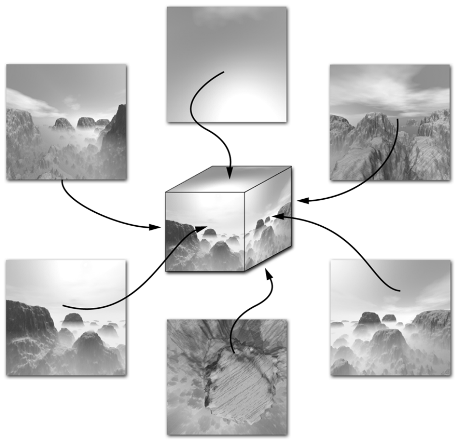
- Assume the Sun is a diffuse emitter of total power $\Phi = 3.91 \cdot 10^{26}$ W and surface area $A = 6.07 \cdot 10^{18}$ m^2.
- Radiance from the Sun to the Earth: $L = \frac{\Phi}{\pi A} \approx 2.05 \cdot 10^7 \; \frac{\text{W}}{\text{m}^2\ \text{sr}}$.
- Assume the Sun is in zenith and at a distance to the Earth of $r = 1.5 \cdot 10^{11}$ m.
- The solid angle subtended by the Sun as seen from Earth: $\omega = \frac{A_s}{r^2} = \frac{A}{4r^2} \approx 6.74 \cdot 10^{-5}$ sr.
- Energy received in a 1×1 m^2 patch of Earth atmosphere: $E = L\omega \approx 1383 \; \frac{\text{W}}{\text{m}^2}$.
- A directional source with its color set to the solar irradiance at Earth will deliver the same energy in a square meter.

Skylight

- Environment mapping:

  Map an omnidirectional image onto everything surrounding the scene.
- Cube mapping:

  Use a direction to perform look-ups into an omnidirectional image consisting of six texture images (square resolution, 90° field of view).
- We use Preetham's analytical model to precompute a sky cube map.
- Look-ups then return the radiance $L_{\text{sky}}(\vec{\omega})$ received from the sky when looking in the direction $\vec{\omega}$.



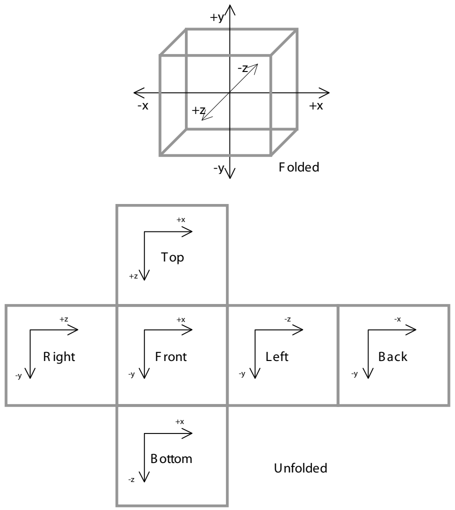
Precomputing a cube map

We precompute by storing $L_{\text{sky}}(\vec{v}_{ijk})$ in the texel at index i, j of face k.

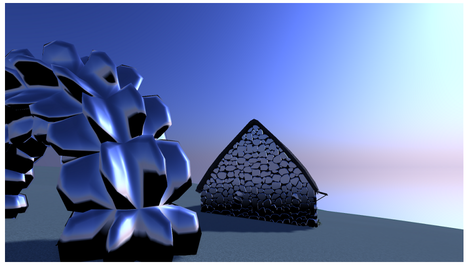
$$\vec{v}_{ijk} = \vec{a}_k + \vec{u}_k \left(2\frac{i}{\text{res}} - 1\right) + \vec{b}_k \left(2\frac{j}{\text{res}} - 1\right),$$

where the faces are numbered from 0 to 5 and ordered as follows: left, right, top, bottom, front, back

- $\vec{a}_k$ is the major axis direction of face k,
- $\vec{u}_k$ is the up direction of face k, and
- $\vec{b}_k$ is the right direction of face k.



Environment mapping



- Reflective environment mapping:

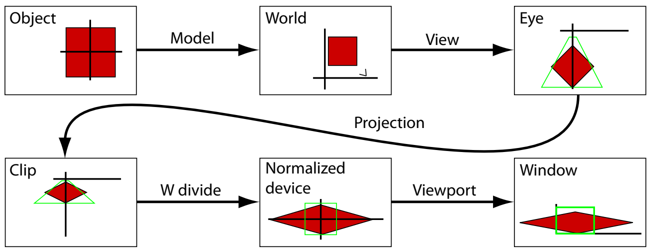
$$\vec{i} = \frac{\boldsymbol{p}_e}{\|\boldsymbol{p}_e\|} \quad , \quad \vec{r} = \vec{i} - 2\left(\vec{n}\cdot\vec{i}\right)\vec{n}.$$

- We need world space directions when looking up in a cube map.

$$\vec{r}_w = \left(\boldsymbol{V}^{-1}\right)^{3\times 3}\vec{r},$$

where $\boldsymbol{V}$ is the view matrix and $\boldsymbol{A}^{3\times 3}$ takes the upper left 3×3 part of $\boldsymbol{A}$.

Environment mapping - filling the background



- Given window space pixel coordinates x_p, y_p find the direction from the eye to the corresponding point on the image plane.

$$\vec{i}_w = \begin{bmatrix} \left(\boldsymbol{V}^{-1}\right)^{3\times 3} & \begin{matrix}0\\0\\0\end{matrix} \\ \begin{matrix}0 & 0 & 0\end{matrix} & 0 \end{bmatrix} \boldsymbol{P}^{-1}\boldsymbol{p}_n \quad , \quad \boldsymbol{p}_n = \begin{bmatrix} 2\frac{x_p}{W} - 1 \\ 2\frac{y_p}{H} - 1 \\ 0 \\ 1 \end{bmatrix}.$$

- Note that depth is unimportant as $\vec{i}_w$ is the direction of an eye ray.

The Rendering Equation

- When rendering surfaces, the equation we try to evaluate is [Kajiya 1986]

$$L_o(\boldsymbol{x},\vec{\omega}) = L_e(\boldsymbol{x},\vec{\omega}) + \int_{2\pi} f_r(\boldsymbol{x},\vec{\omega}',\vec{\omega}) L_i(\boldsymbol{x},\vec{\omega}') \cos\theta \, d\omega' \ ,$$

- where
  - L_o is outgoing radiance,
  - L_e is emitted radiance,
  - L_i is incoming radiance,
  - $\boldsymbol{x}$ is a surface position,
  - $\vec{\omega}$ is the direction of the light,
  - $\vec{\omega}'$ is the direction toward the light source,
  - f_r is the bidirectional reflectance distribution function (BRDF),
  - $d\omega'$ is an element of solid angle,
  - θ is the angle between $\vec{\omega}'$ and the surface normal $\vec{n}$ at $\boldsymbol{x}$, such that $\cos\theta = \vec{\omega}' \cdot \vec{n}$.

References

- Kajiya, J. The Rendering Equation. *Computer Graphics (Proceedings of ACM SIGGRAPH 86) 20*(4), pp. 143–150, 1986.

Splitting the evaluation

- Distinguishing between:
  - Direct illumination L_{direct}.
    - Light reaching a surface directly from the source.
  - Indirect illumination L_{indirect}.
    - Light reaching a surface after at least one bounce.
- The rendering equation is then

$$L = L_e + L_{\text{direct}} + L_{\text{indirect}} \, .$$

  - L_e is emission.
  - L_{direct} is sampling of lights.
  - L_{indirect} is sampling of the BRDF excluding lights.
- In a real-time skylight setting:
  - L_{direct} is direct sunlight (using the Phong illumination model without ambient).
  - L_{indirect} is integrated skylight and specular reflection of the sky.

Sky irradiance E_{sky} and ambient occlusion A

- Excluding direct sunlight from L_i, we have

$$\begin{aligned} L_{\text{indirect}}(\boldsymbol{x}, \vec{\omega}_o) &= \int_{2\pi} f_r(\boldsymbol{x}, \vec{\omega}', \vec{\omega}) L_i(\boldsymbol{x}, \vec{\omega}_i)(\vec{n} \cdot \vec{\omega}_i) \, d\omega_i \\ &= \frac{\rho_d}{\pi} \int_{2\pi} V(\boldsymbol{x}, \vec{\omega}_i) L_{\text{sky}}(\vec{\omega}_i)(\vec{n} \cdot \vec{\omega}_i) \, d\omega_i + \rho_s V(\boldsymbol{x}, \vec{r}) L_{\text{sky}}(\vec{r}_w) \, , \end{aligned}$$

  where
  - $L_i(\boldsymbol{x}, \vec{\omega}_i) = V(\boldsymbol{x}, \vec{\omega}_i) L_{\text{sky}}(\vec{\omega}_i)$ is incident skylight,
  - $f_r(\boldsymbol{x}, \vec{\omega}', \vec{\omega}) = \rho_d / \pi$ is the BRDF of perfectly diffuse (Lambertian) materials,
  - ρ_d is the diffuse reflectance (diffuse color), and
  - ρ_s is the specular reflectance.
- The remaining integral is called the *irradiance* E, and

$$\begin{aligned} E(\boldsymbol{x}, \vec{n}) &= \int_{2\pi} V(\boldsymbol{x}, \vec{\omega}_i) L_{\text{sky}}(\vec{\omega}_i)(\vec{n} \cdot \vec{\omega}_i) \, d\omega_i \\ &\approx \frac{1}{\pi} \int_{2\pi} V(\boldsymbol{x}, \vec{\omega}_i)(\vec{n} \cdot \vec{\omega}_i) \, d\omega_i \int_{\Omega} L_{\text{sky}}(\vec{\omega}_i)(\vec{n} \cdot \vec{\omega}_i) \, d\omega_i \\ &= A(\boldsymbol{x}) E_{\text{sky}}(\vec{n}) \, . \end{aligned}$$

Integrating skylight

- The sky irradiance integral (neglecting visibility):

$$E_{\text{sky}}(\vec{n}) = \int_{2\pi} L_{\text{sky}}(\vec{\omega}_i)(\vec{n} \cdot \vec{\omega}_i) \, d\omega_i \, .$$

- Monte Carlo estimator:

$$E_{\text{sky},N}(\vec{n}) = \frac{1}{N} \sum_{j=1}^{N} \frac{L_{\text{sky}}(\vec{\omega}_{i,j})(\vec{n} \cdot \vec{\omega}_{i,j})}{\text{pdf}(\vec{\omega}_{i,j})} \, .$$

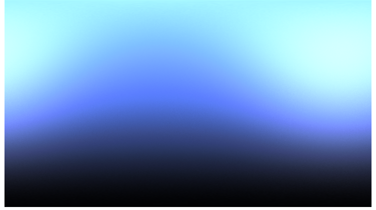
- A good choice of pdf would be $\text{pdf}(\vec{\omega}_{i,j}) = \frac{\vec{n} \cdot \vec{\omega}_{i,j}}{\pi}$.
- We sample $\vec{\omega}_{i,j}$ on a cosine-weighted hemisphere using

$$\begin{aligned} (\theta_i, \phi_i) &= (\cos^{-1}\sqrt{1 - \xi_1}, 2\pi\xi_2) \quad , \quad \xi_1, \xi_2 \in [0, 1) \quad \text{(random numbers)} \\ (x, y, z) &= (\cos\phi_i \sin\theta_i, \sin\phi_i \sin\theta_i, \cos\theta_i) \\ \vec{\omega}_{i,j} &= \left(x \begin{bmatrix} 1 - n_x^2/(1 + n_z) \\ -n_x n_y/(1 + n_z) \\ -n_x \end{bmatrix} + y \begin{bmatrix} -n_x n_y/(1 + n_z) \\ 1 - n_y^2/(1 + n_z) \\ -n_y \end{bmatrix} + z \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \right) . \end{aligned}$$

Irradiance environment map

skylight

to



sky irradiance

- Precompute cube map on the GPU using layered rendering and sampling.
- In the geometry shader (integrator.geom):
  - emit triangle vertices to each cube map face using a loop and `gl_Layer`.
- In the fragment shader (integrator.frag):
  - implement the cosine-weighted sampling of the skylight with $\vec{n} = \vec{v}_{ijk}$ (use `rnd(t)` to get a random number).
- The framework progressively improves the sky irradiance map using these shaders.

Illumination model for skylight

$$L_r = \frac{\rho_d}{\pi}\left((\vec{n}\cdot\vec{l})V\,L_{\text{sun}} + A\,E_{\text{sky}}(\vec{n}_w)\right) + \rho_s\left((\vec{r}\cdot\vec{l})^p V\,L_{\text{sun}} + L_{\text{sky}}(\vec{r}_w)\right),$$

where

- L_{sun} is the radiance (color) of the directional light,
- L_{sky} is a look-up into the skylight environment map,
- E_{sky} is a look-up into the sky irradiance map,
- V is the visibility term obtained from shadow mapping,
- A is the factor from screen-space ambient occlusion (SSAO),
- $\vec{n}$ is the normal,
- $\vec{l}$ is the direction toward the sun,
- $\vec{r}$ is the reflection of the view vector around the normal,
- $\vec{n}_w$ is the normal in world coordinates,
- $\vec{r}_w$ is the reflected vector in world coordinates,
- ρ_d is a diffuse reflectance,
- ρ_s is a specular reflectance, and
- p is a shininess.