Volume Rendering 方程式から大気散乱モデルを導く

2018.9.1

レイトレ合宿6

@ishiyama_

* このスライドは，レイトレ合宿6アドベントカレンダーの内容に基づいています：
https://github.com/k-ishiyama/rt6_advent_calendar5/blob/master/doc/atmospheric_scattering_model.pdf

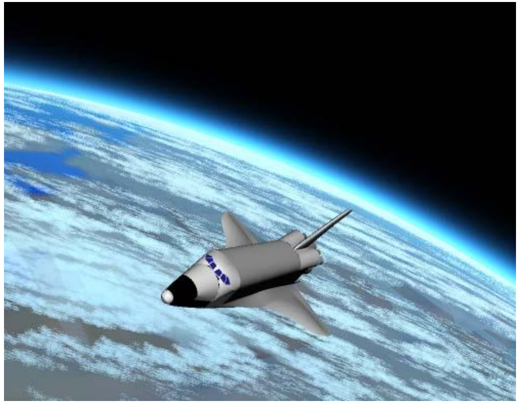
大気散乱モデル

地球大気をモデル化 [NSTN93]

- 光源は太陽のみ
- レイリー散乱、ミー散乱
- 高さ方向に媒質濃度が減衰

事前計算大気散乱 [BN08, EK10]

- テクスチャに事前計算
- 多重散乱



[NSTN93]

[EK10]

動機

大気散乱モデルで作った天球を IBL に使う

→ レイトレーシングのときに散乱係数を流用しよう

→ Volume Rendering方程式との関係を確かめておく

内容

Volume Rendering 方程式から出発して,

[EK10] Oskar Elek and Petr Kmoch. Real-Time Spectral Scattering in Large-Scale Natural Participating Media.

の内部散乱項

$$I^{(1)}_{S_{\{R,M\}}}(P_O,V,L,\lambda) = R(P_O,V,L,\lambda) + \boxed{I_I(\lambda)F_{\{R,M\}}(\theta) \cdot \frac{\sigma_{\{R,M\}}(\lambda)}{4\pi} \int_{P_a}^{P_b} \rho_{\{R,M\}}(P) e^{-t(PP_c,\lambda)-t(P_aP,\lambda)} ds} \quad (7)$$

$$I^{(k)}_{S_{\{R,M\}}}(P_O,V,L,\lambda) = A^{(k-1)}_{\{R,M\}}(P_O,V,L,\lambda) + \boxed{\frac{\sigma_{\{R,M\}}(\lambda)}{4\pi} \cdot \int_{P_a}^{P_b} I^{(k-1)}_{G_{\{R,M\}}}(P,V,L,\lambda)\rho_{\{R,M\}}(P) e^{-t(P_aP,\lambda)} ds} \quad (12)$$

の導出に焦点を当てる.

内容

- 大気散乱モデル
- Volume Rendering 方程式のおさらい
- 大気散乱モデルの内部散乱項の導出
  - 大気散乱モデル: 内部散乱項 (一般)
  - 大気散乱モデル: 内部散乱項 (地球)
- まとめ

Volume Rendering 方程式のおさらい

Radiative transfer 方程式[Cha60] (Notation は [NGHJ18])

$$(\boldsymbol{\omega} \cdot \nabla) L(\boldsymbol{x}, \boldsymbol{\omega}) = -\mu_e(\boldsymbol{x}) L(\boldsymbol{x}, \boldsymbol{\omega}) + \mu_a(\boldsymbol{x}) L_e(\boldsymbol{x}, \boldsymbol{\omega}) + \mu_s(\boldsymbol{x}) L_s(\boldsymbol{x}, \boldsymbol{\omega})$$

$$L_s(\boldsymbol{x}, \boldsymbol{\omega}) = \oint p(\boldsymbol{\omega}, \boldsymbol{\omega}') L_i(\boldsymbol{x}, \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$

- x ：位置，ω : 方向
- L : 光線の輝度、Le : 自己発光輝度、Li : 入射輝度
- μa: 吸収係数、μs: 散乱係数、μe : 消散係数
- p : 散乱の位相関数

Volume Rendering 方程式のおさらい

Volume Rendering 方程式 (Radiative Transfer 方程式の積分形)

$$
\begin{aligned}
L(\boldsymbol{x}(s),\, \boldsymbol{\omega}) = T(\boldsymbol{x}(0) \to \boldsymbol{x}(s)) L\left(\boldsymbol{x}(0)\right) \\
+ \int_0^s T(\boldsymbol{x}(s') \to \boldsymbol{x}(s)) \left[\mu_a\left(\boldsymbol{x}(s)\right) L_e\left(\boldsymbol{x}(s),\, \boldsymbol{\omega}\right) + \mu_s\left(\boldsymbol{x}(s')\right) L_s\left(\boldsymbol{x}(s'),\, \boldsymbol{\omega}\right)\right] ds'
\end{aligned}
$$

$$
T(\boldsymbol{x}(a) \to \boldsymbol{x}(b)) = \exp\left(-\int_a^b \mu_e\left(\boldsymbol{x}(t)\right) dt\right)
$$

ボリュームレンダリングの基礎となる方程式.

ここから[EK10]の内部散乱項を導く.

大気散乱モデルの導出

- 大気散乱モデル: 内部散乱項 (一般)
- 大気散乱モデル: 内部散乱項 (地球)

流れ

Volume Rendering 方程式

複数の散乱媒質 , 光源は太陽 etc.

大気散乱モデル: 内部散乱項(一般)

レイリー散乱、ミー散乱
高さに応じて大気濃度が指数減衰

大気散乱モデル:内部散乱項(地球)

Volume Rendering 方程式に対する条件

基本的に [NSTN93] を踏襲

- 自己発光なし

$$L_e\left(\boldsymbol{x},\, \boldsymbol{\omega}\right) = 0$$

- 複数の散乱媒質

$$\mu_s\left(\boldsymbol{x}\right) = \sum_j \mu_{sj}\left(\boldsymbol{x}\right), \quad \mu_a\left(\boldsymbol{x}\right) = \sum_j \mu_{aj}\left(\boldsymbol{x}\right)$$

Volume Rendering 方程式に対する条件

- 光源は太陽光のみ. 太陽光は平行光として近似

$$L_i(\boldsymbol{x}, \boldsymbol{\omega}) = L_{\mathrm{sun}}\delta(\boldsymbol{\omega} - \boldsymbol{\omega}_{\mathrm{sun}})$$

- レイは直進
  - 積分を簡単にするため

- 宇宙と大気、大気と地表を隔てる境界がある
  - 積分範囲を決めるため

Volume Rendering 方程式に対する条件

- 入射光は前回の散乱結果にのみ依存

$$L_s^{(n)}(\boldsymbol{x}(s')) = \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}')L^{(n-1)}(\boldsymbol{x}(s'), \boldsymbol{\omega}')d\Omega(\boldsymbol{\omega}')$$

Volume Rendering 方程式を漸化式として書く

$$L^{(n)}(\boldsymbol{x}(s)) = T(\boldsymbol{x}(0) \to \boldsymbol{x}(s))L^{(n)}(\boldsymbol{x}(0)) + \int_0^s T(\boldsymbol{x}(s') \to \boldsymbol{x}(s))\mu_s(\boldsymbol{x}(s'))L_s^{(n)}(\boldsymbol{x}(s'))ds'$$

以降, 内部散乱項に注目

$$L_{\text{inscatter}}^{(n)}(\boldsymbol{x}(s)) = \int_0^s T(\boldsymbol{x}(s') \to \boldsymbol{x}(s))\mu_s(\boldsymbol{x}(s'))L_s^{(n)}(\boldsymbol{x}(s'))ds'$$

Volume Rendering 方程式に対する条件

- 散乱媒質間の相互作用なし[EK10]

$$L_{\text{inscatter}}^{(n)} = \sum_n \left(L_R^{(n)} + L_M^{(n)} \right)$$

求める内部散乱項

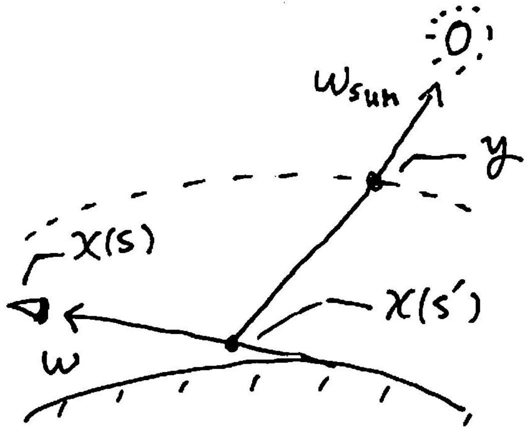
$$L_{\text{inscatter}}^{(n)}(\boldsymbol{x}(s)) = \sum_j \int_0^s T(\boldsymbol{x}(s') \to \boldsymbol{x}(s))\mu_{sj}(\boldsymbol{x}(s'))L_{sj}^{(n)}(\boldsymbol{x}(s'))ds'$$

$$L_{sj}^{(n)}(\boldsymbol{x}(s')) = \oint p_j(\boldsymbol{\omega},\, \boldsymbol{\omega}')L_j^{(n-1)}(\boldsymbol{x}(s'),\, \boldsymbol{\omega}')d\Omega(\boldsymbol{\omega}')$$

1次散乱

入射光：太陽

$$L_j^{(0)}(\boldsymbol{x}(s'),\, \boldsymbol{\omega}') = T(\boldsymbol{y} \to \boldsymbol{x}(s'))L_{\mathrm{sun}}\delta(\boldsymbol{\omega}' - \boldsymbol{\omega}_{\mathrm{sun}})$$



散乱光

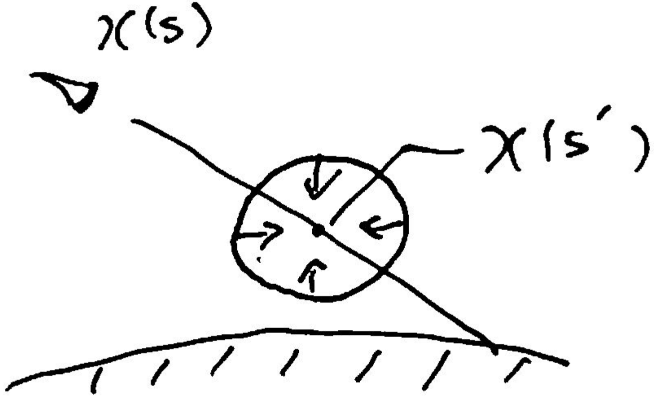
$$L_{sj}^{(1)}(\boldsymbol{x}(s')) = \oint p_j(\boldsymbol{\omega},\, \boldsymbol{\omega}')T(\boldsymbol{y} \to \boldsymbol{x}(s'))L_{\mathrm{sun}}\delta(\boldsymbol{\omega}' - \boldsymbol{\omega}_{\mathrm{sun}})d\Omega(\boldsymbol{\omega}')$$

$$= p_j(\boldsymbol{\omega},\, \boldsymbol{\omega}_{\mathrm{sun}})L_{\mathrm{sun}}T(\boldsymbol{y} \to \boldsymbol{x}(s'))$$

高次散乱(n>1)

入射光：前回の散乱結果

$$L_j^{(n-1)}(\boldsymbol{x}(s'),\, \boldsymbol{\omega}')$$



散乱光

$$L_{sj}^{(n)}(\boldsymbol{x}(s')) = \oint p_j(\boldsymbol{\omega},\, \boldsymbol{\omega}') L_j^{(n-1)}(\boldsymbol{x}(s'),\, \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$

大気散乱モデル: 内部散乱項 (一般)

$$L^{(n)}_{\text{inscatter}}(\boldsymbol{x}(s)) = \sum_j \int_0^s T(\boldsymbol{x}(s') \to \boldsymbol{x}(s)) \mu_{sj}(\boldsymbol{x}(s')) L^{(n)}_{sj}(\boldsymbol{x}(s')) ds'$$

Transmittance

$$T(\boldsymbol{x}(a) \to \boldsymbol{x}(b)) = \exp\left(-\sum_j \int_a^b \mu_{ej}\left(\boldsymbol{x}(t)\right) dt \right)$$

1次散乱

$$L^{(1)}_{sj}(\boldsymbol{x}(s')) = p_j(\boldsymbol{\omega}, \boldsymbol{\omega}_{\text{sun}}) L_{\text{sun}} T(\boldsymbol{y} \to \boldsymbol{x}(s'))$$

高次散乱(n>1)

$$L^{(n)}_{sj}(\boldsymbol{x}(s')) = \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}') L^{(n-1)}_j(\boldsymbol{x}(s'), \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$

大気散乱モデル (地球)

あとは, 散乱媒質 を与えれば, 具体的な表現が得られる.

[NSTN93, BN08, EK10] と同様に

- 大気 : レイリー散乱とミー散乱、高度に従って指数関数で減衰

とする

散乱媒質に対する条件

散乱媒質

- レイリー散乱、ミー散乱

$$p_R = \frac{3}{16\pi}(1+\cos^2\theta) \qquad p_M = \frac{3}{8\pi}\frac{1-g^2}{2+g^2}\frac{1+\cos^2\theta}{(1+g^2-2g\cos\theta)^{3/2}}$$

※ $\theta = \cos^{-1}(\boldsymbol{\omega}\cdot\boldsymbol{\omega}')$

- 高さ h(x) に従って減衰

$$\sum_j \mu_{aj}(\boldsymbol{x}) = \sum_j^{R,\,M} \beta_{aj} e^{-\frac{h(\boldsymbol{x})}{H_j}}, \quad \sum_j \mu_{sj}(\boldsymbol{x}) = \sum_j^{R,\,M} \beta_{sj} e^{-\frac{h(\boldsymbol{x})}{H_j}}$$

大気散乱モデル(地球): 内部散乱項

$$L^{(n)}_{\text{inscatter}}(\boldsymbol{x}(s)) = \sum_{j}^{R,\,M} \int_0^s T(\boldsymbol{x}(s') \to \boldsymbol{x}(s))\mu_{sj}(\boldsymbol{x}(s'))L^{(n)}_{sj}(\boldsymbol{x}(s'))ds'$$

Transmittance

$$T(\boldsymbol{x}(a) \to \boldsymbol{x}(b)) = \exp\left(-\sum_{j}^{R,\,M} \beta_{ej} \int_a^b e^{-\frac{h(\boldsymbol{x}(t))}{H_j}}\,(\boldsymbol{x}(t))\,dt\right)$$

1次散乱

$$L^{(1)}_{sj}(\boldsymbol{x}(s')) = p_j(\boldsymbol{\omega},\,\boldsymbol{\omega}_{\text{sun}})L_{\text{sun}}T(\boldsymbol{y} \to \boldsymbol{x}(s'))$$

高次散乱(n>1)

$$L^{(n)}_{sj}(\boldsymbol{x}(s')) = \oint p_j(\boldsymbol{\omega},\,\boldsymbol{\omega}')L^{(n-1)}_{j}(\boldsymbol{x}(s'),\,\boldsymbol{\omega}')d\Omega(\boldsymbol{\omega}')$$

まとめ

Volume Rendering方程式から出発して

- [NSTN93] の条件

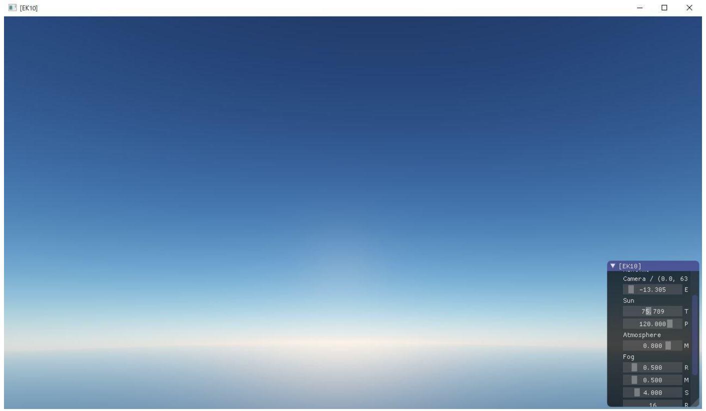
そして

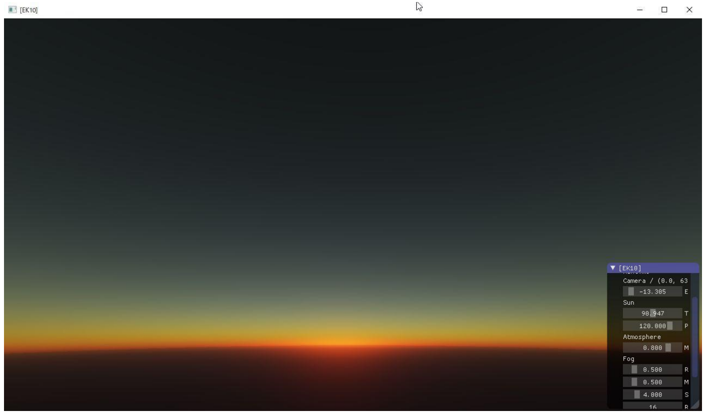
- [EK10] の導出過程

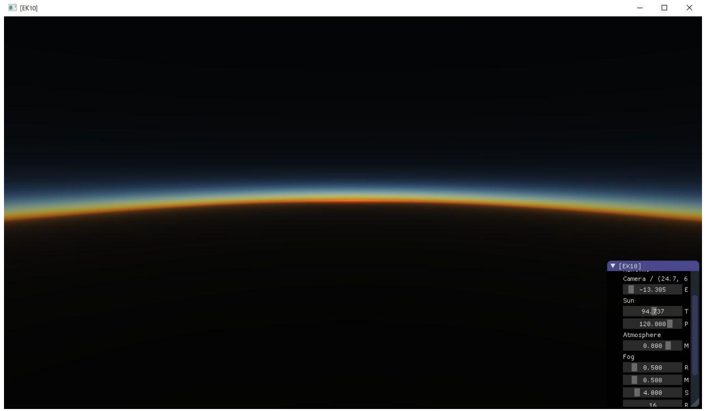
を追うことで [EK10]と同様の内部散乱項を導き,

大気散乱モデルは Volume Rendering 方程式のひとつの形であることを確かめた

実装結果：内部散乱項のみ







参考文献

[Cha60] Subrahmanyan Chandrasekhar. *Radiative Transfer*. Dover Publications Inc., 1960

[Kaj86] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph*., 20(4):143--150, August 1986.

[NSTN93] Tomoyuki Nishita, Takao Sirai, Katsumi Tadamura, and Eihachiro Nakamae. Display of The Earth Taking into Account Atmospheric Scattering. In *Proceedings of the 20th Annual Conferene on Computer Graphics and Interactive Tehniques*, SIGGRAPH'93, pages 175--182, New York, NY, USA, 1993. ACM

[BN08] Eric Bruneton and Fabrice Neyret. Precomputed Atmospheric Scattering. *Computer Graphics Forum*, 27(4):1079--1086, 2008.

[EK10] Oskar Elek and Petr Kmoch. Real-Time Spectral Scattering in Large-Scale Natural Participating Media. In *Proceedings of the 26th Spring Conference on Computer Graphics*, SCCG'10, pages 77--84, New York, NY, USA, 2010. ACM

[NGHJ18] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. *Computer Graphics Forum (Proceedings of Eurographics - State of the Art Reports)*, 37(2), may 2018.