GameDevelopers Conference

Rendering Outdoor Light Scattering in Real Time

Naty Hoffman
Westwood Studios
naty@westwood.com

Arcot J Preetham
ATI Research
preetham@ati.com

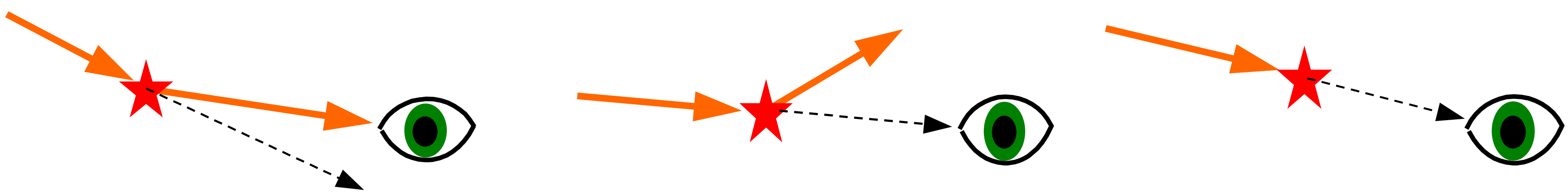
make better games

Outline

- Basics
  - Atmospheric Light Scattering
  - Radiometric Quantities
  - From Radiance to Pixels
- Scattering Theory
  - Absorption, Out-Scattering, In-Scattering
  - Rayleigh and Mie Scattering
- Implementation
  - Aerial Perspective, Sunlight, Skylight
  - Vertex Shader
- Future Work

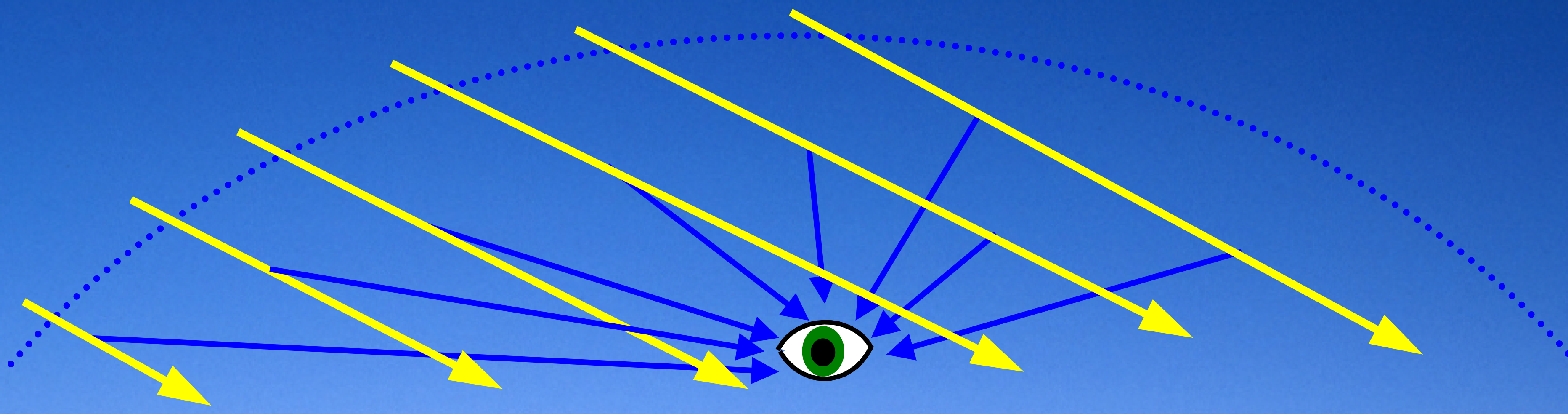
Atmospheric Light Scattering

- Is caused by a variety of particles
  - Molecules, dust, water vapor, etc.
- These can cause light to be:
  - Scattered into the line of sight (in-scattering)
  - Scattered out of the line of sight (out-scattering)
  - Absorbed altogether (absorption)



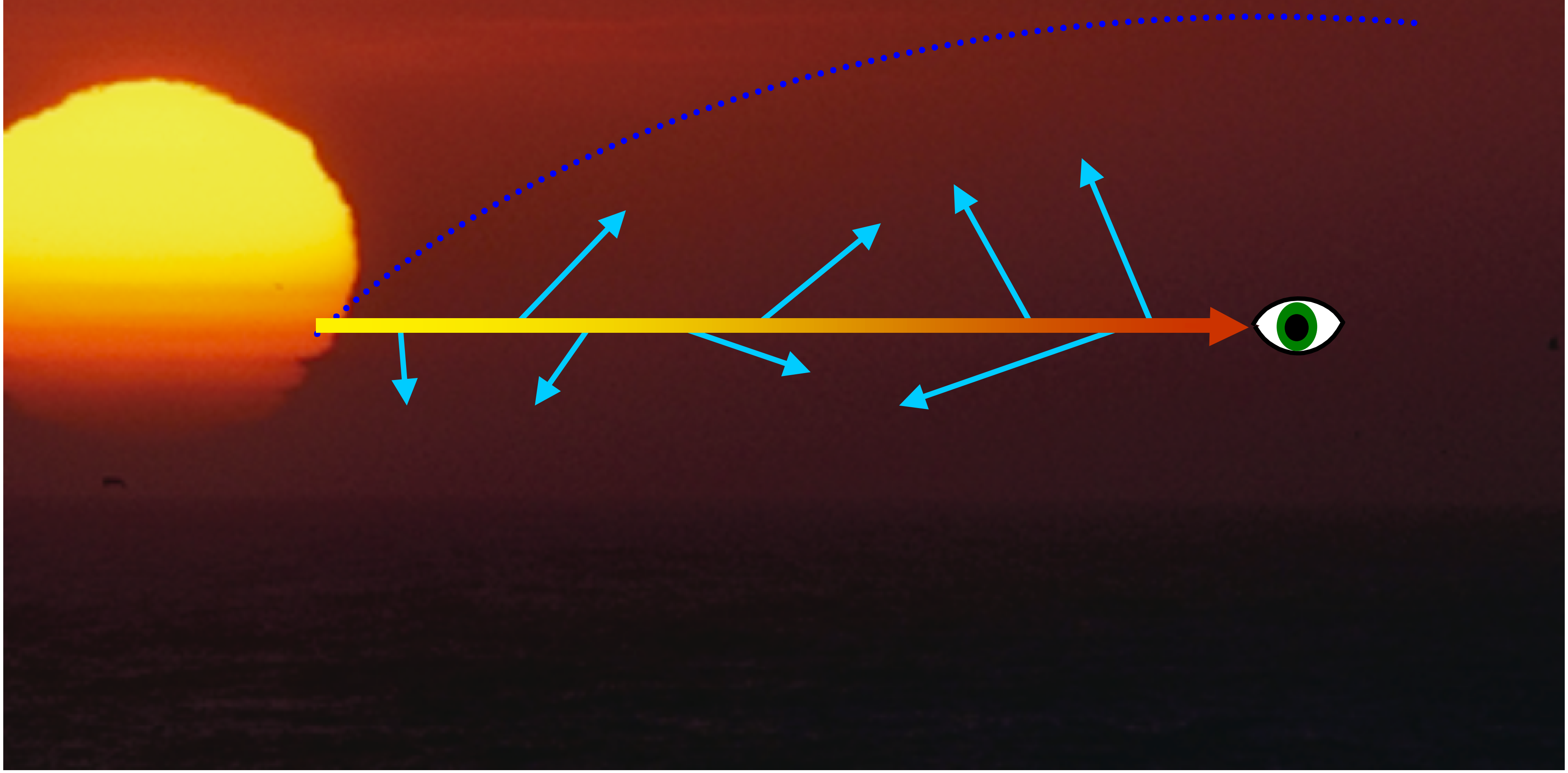
Atmospheric Light Scattering

- Illuminates the sky



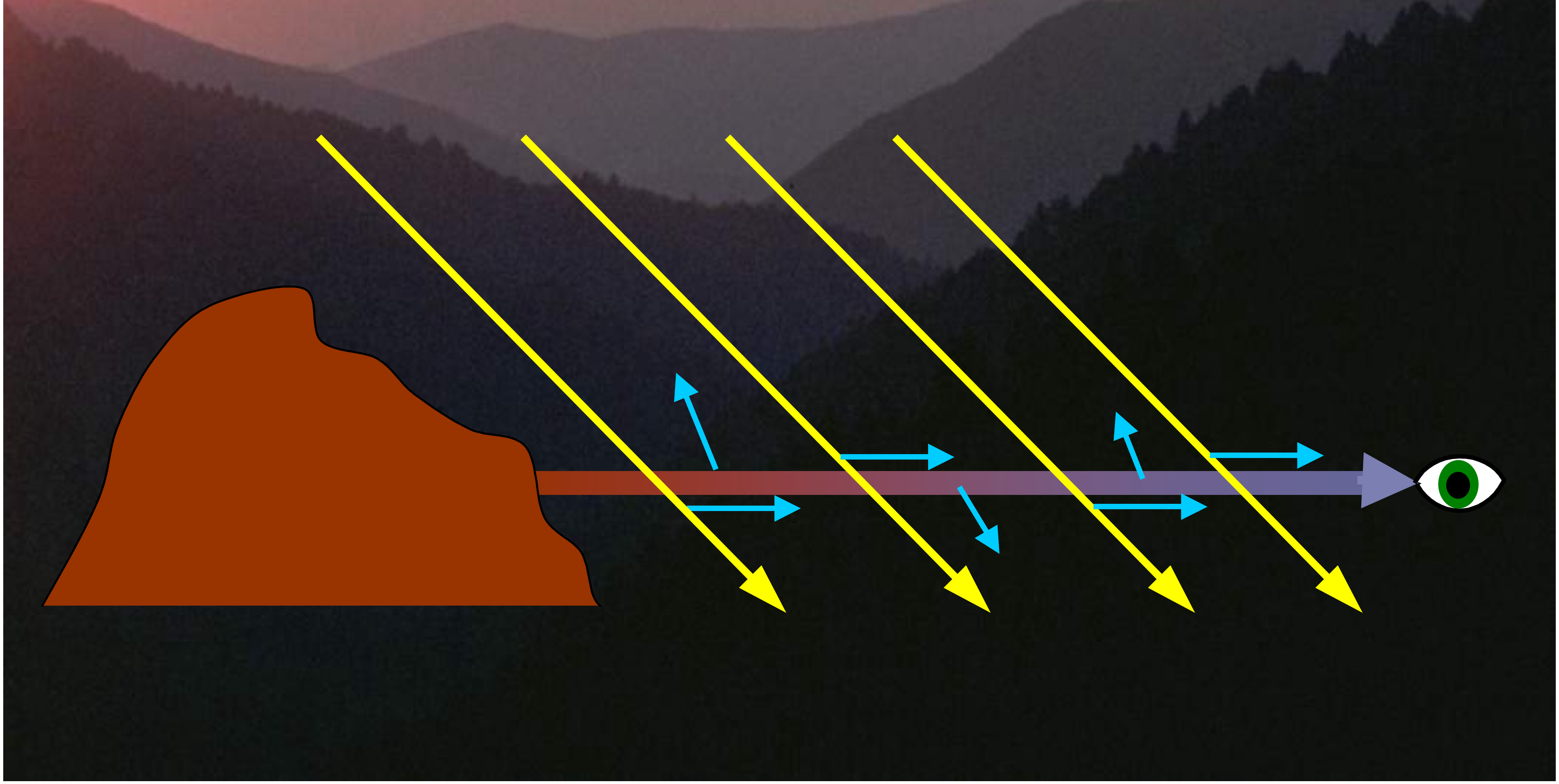
Atmospheric Light Scattering

- Attenuates and colors the Sun



Atmospheric Light Scattering

- Attenuates and colors distant objects

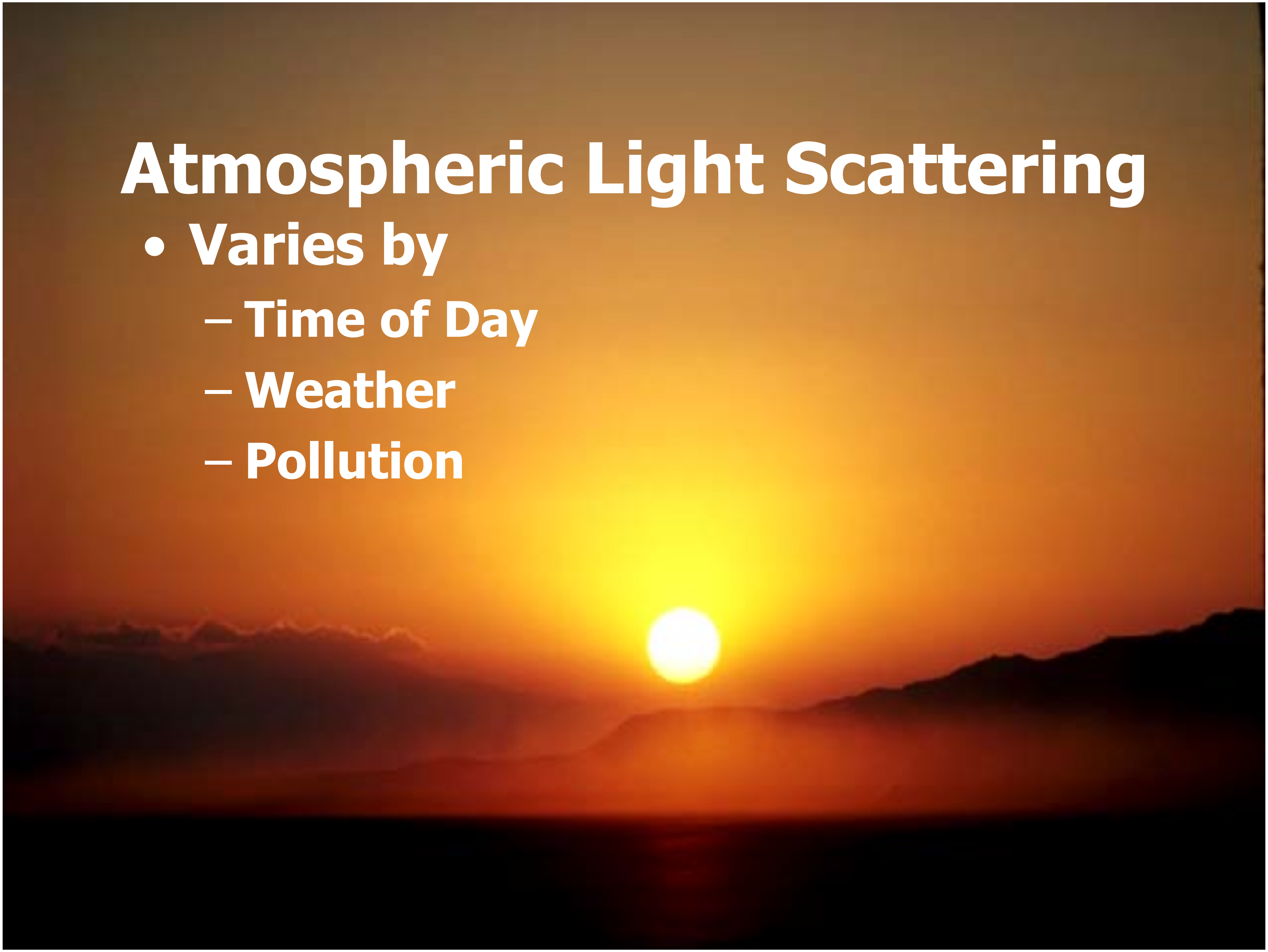


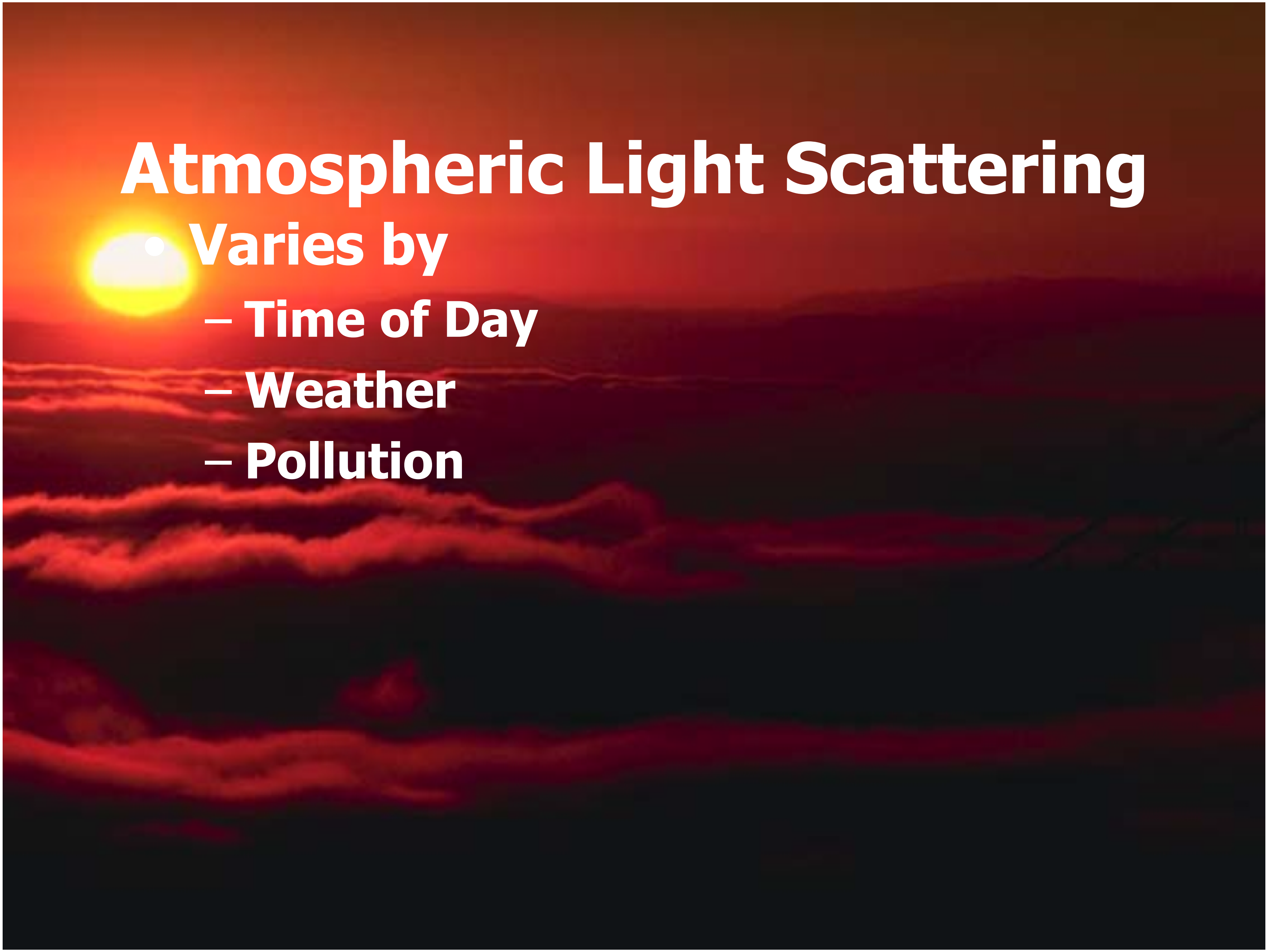
Atmospheric Light Scattering

- Varies by
  - Time of Day
  - Weather
  - Pollution

Atmospheric Light Scattering

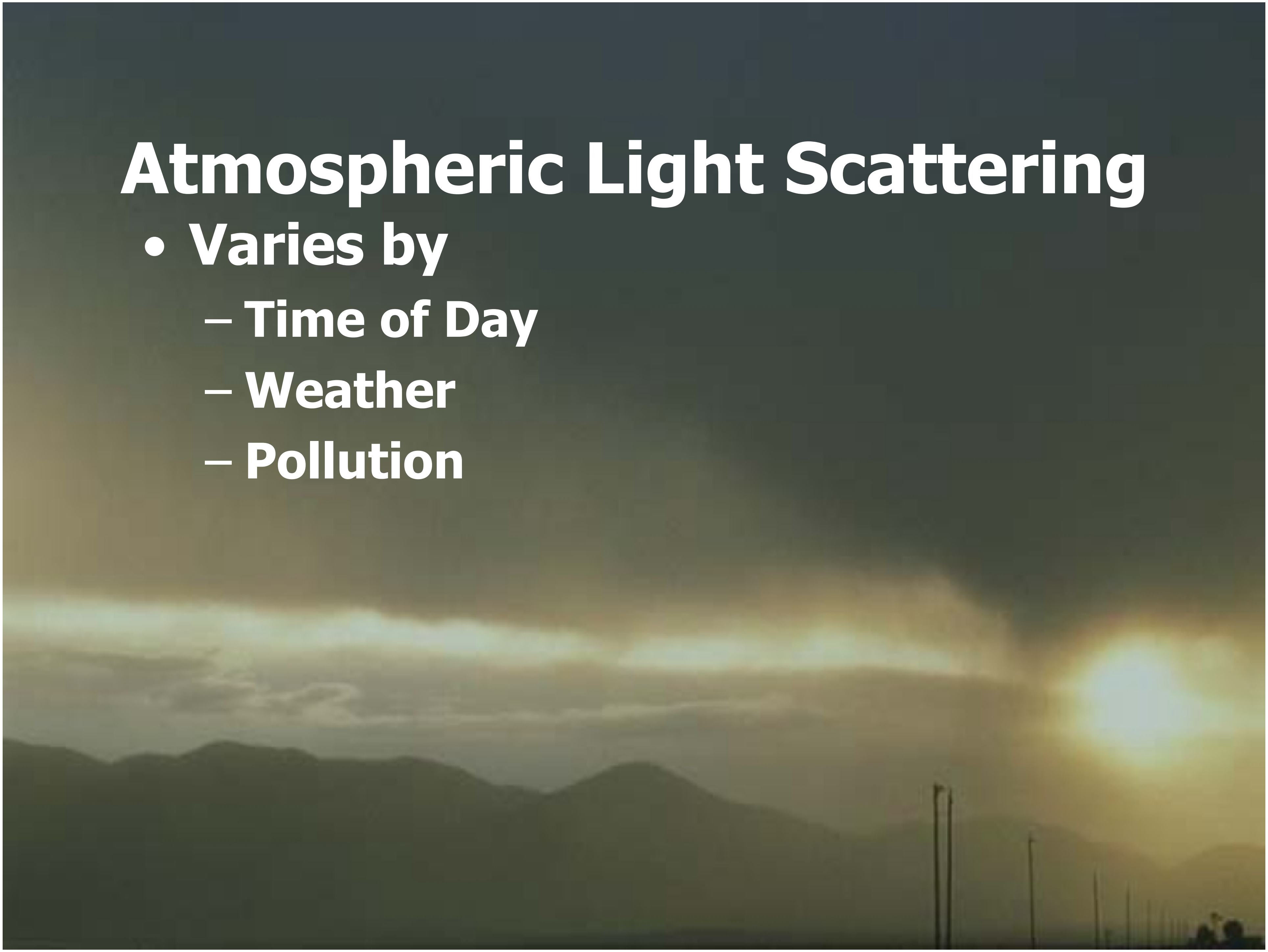
- Varies by
  - Time of Day
  - Weather
  - Pollution




Atmospheric Light Scattering
• Varies by
– Time of Day
– Weather
– Pollution

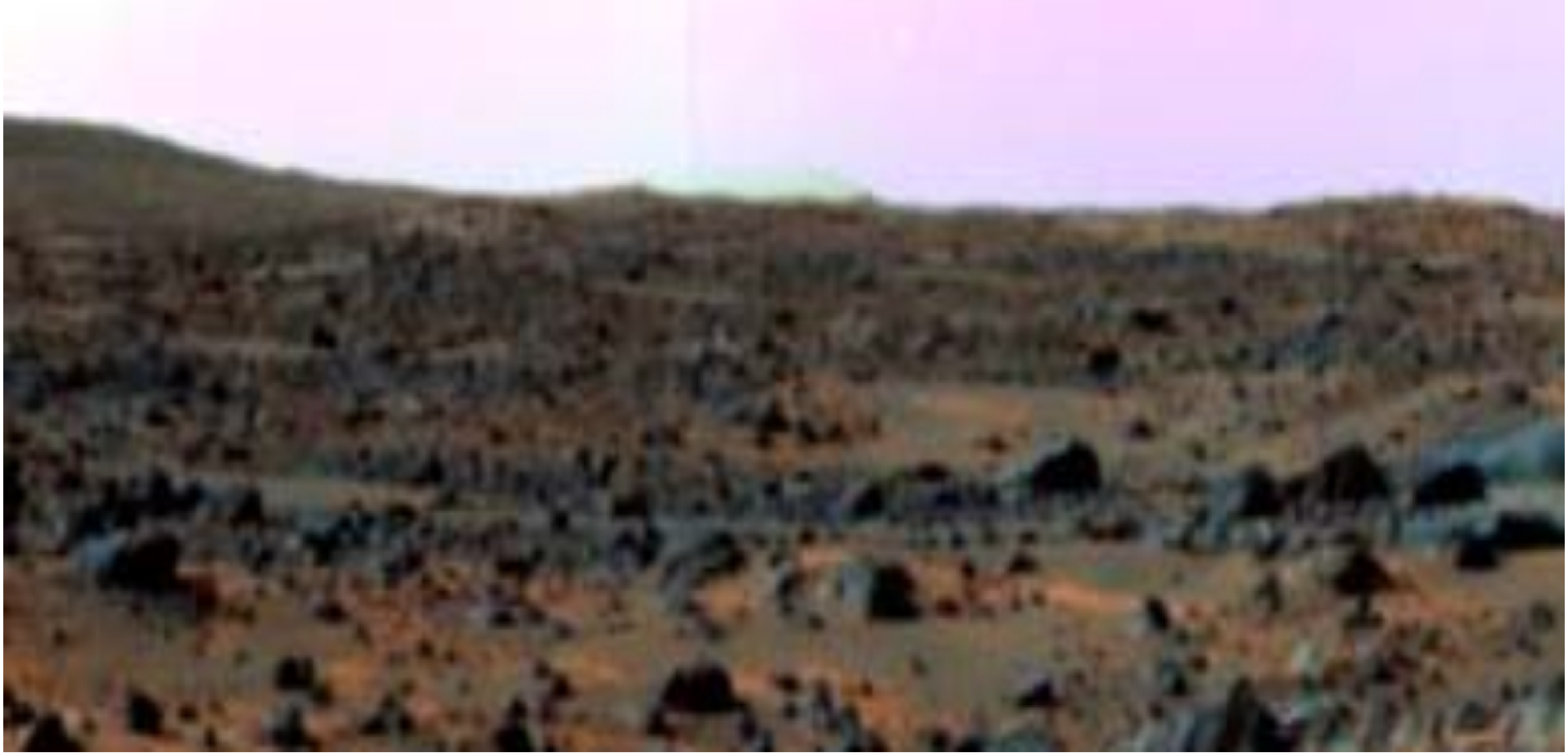
Atmospheric Light Scattering

- Varies by
  - Time of Day
  - Weather
  - Pollution



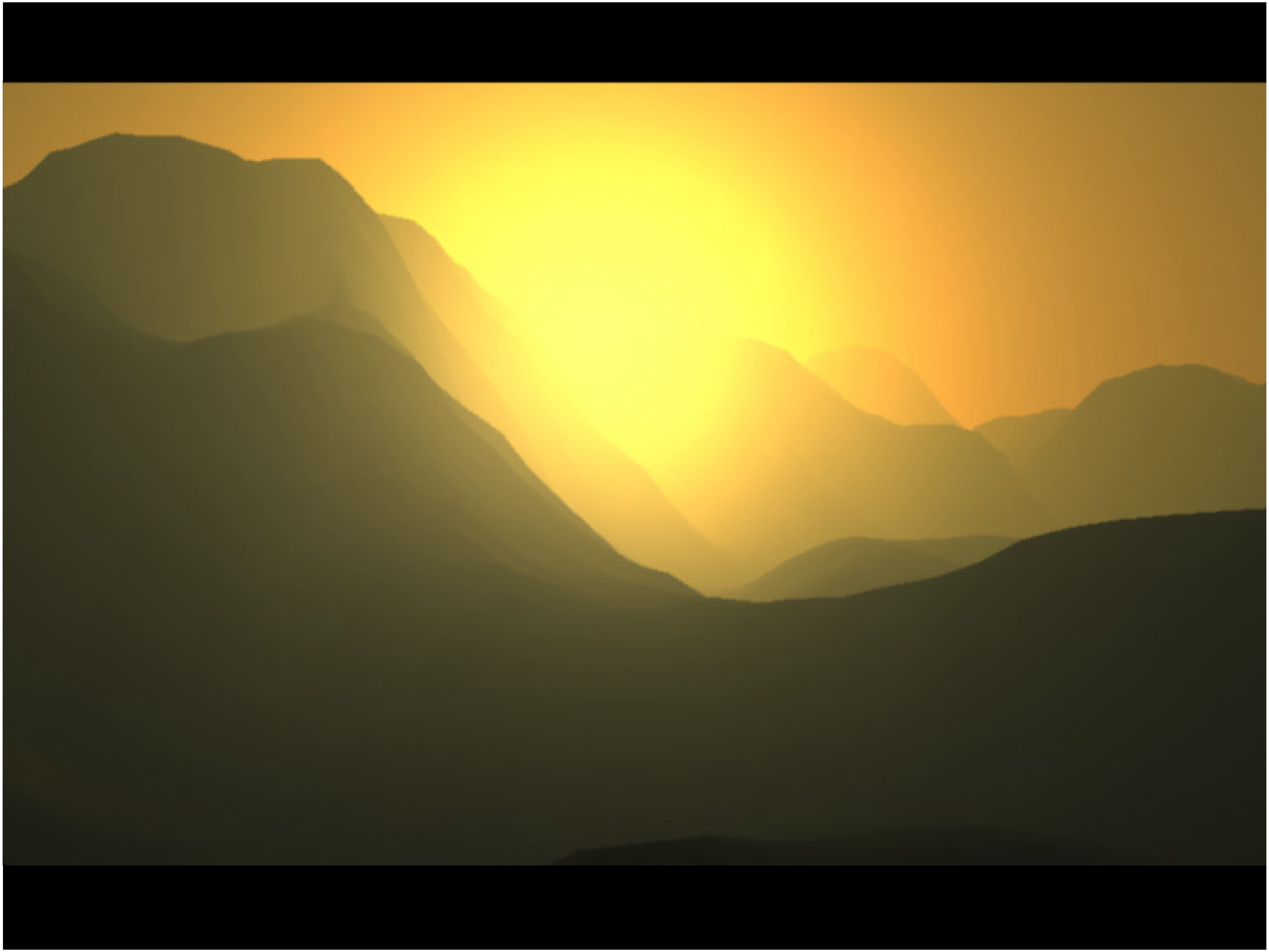
Atmospheric Light Scattering

- **Varies between planets**



Atmospheric Light Scattering

- Extinction (Absorption, Out-scattering)
  - Phenomena which remove light
  - Multiplicative: $L_{\text{extinction}} = F_{\text{ex}} L_0$
- In-scattering:
  - Phenomenon which adds light
  - Additive: L_{in}
- Combined: $L_{\text{scattering}} = F_{\text{ex}} L_0 + L_{\text{in}}$

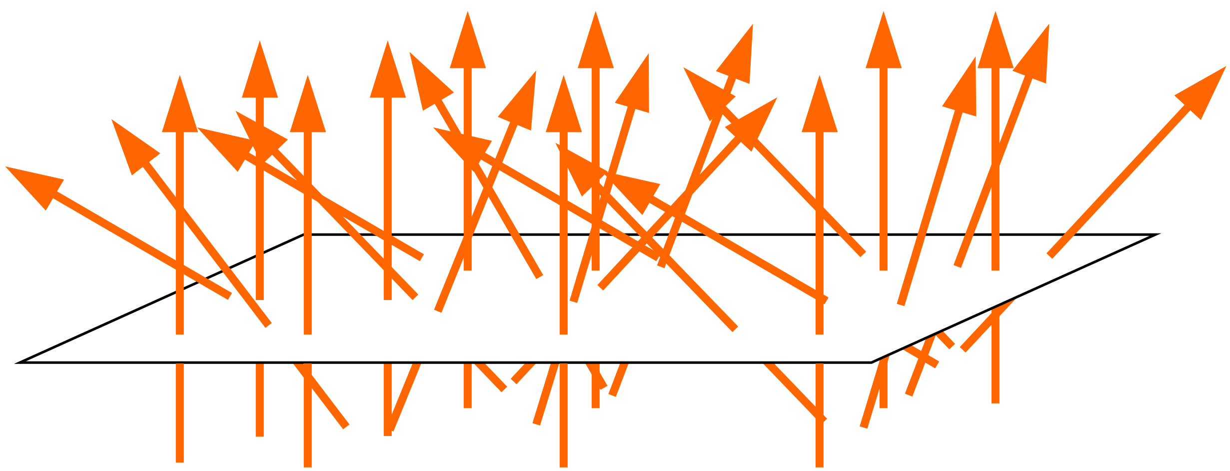


Radiometric Quantities

- Radiant Flux
- Radiance
- Irradiance

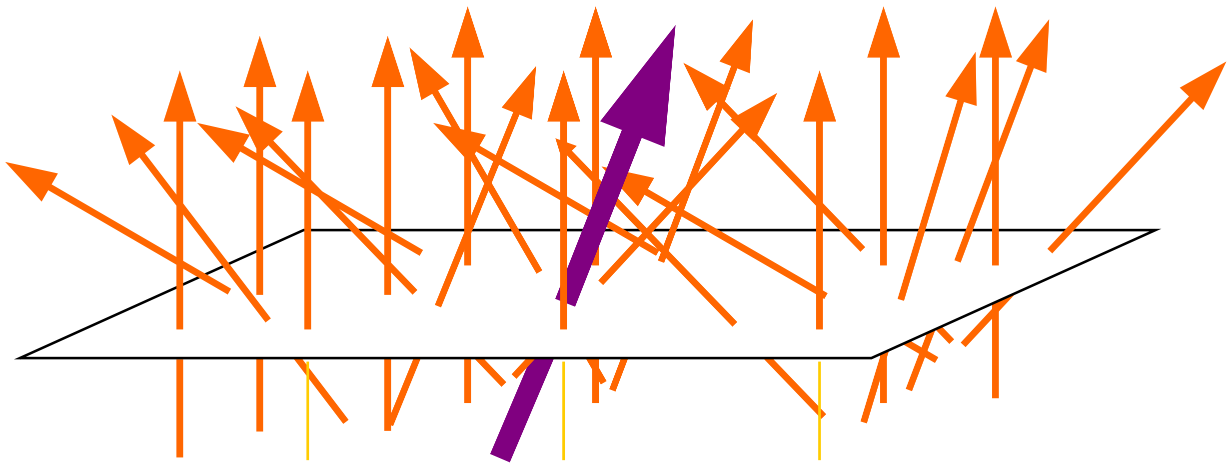
Radiometric Quantities

- Radiant Flux Φ
  - Quantity of light through a surface
  - Radiant power (energy / time)
  - Watt



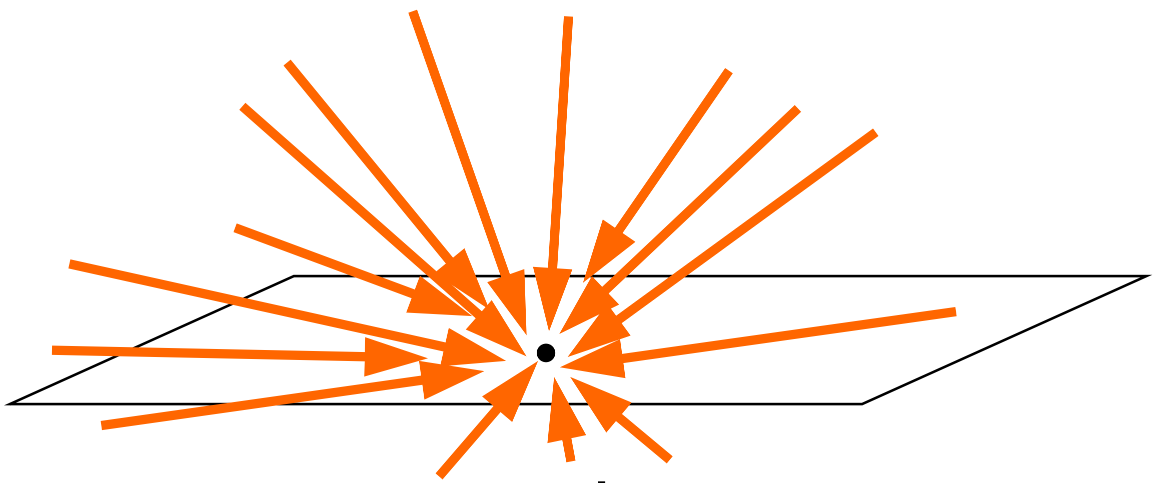
Radiometric Quantities

- **Radiance** L
  - **Quantity of light in a single ray**
  - **Radiant flux / area / solid angle**
  - **Watt / (meter^2 * steradian)**



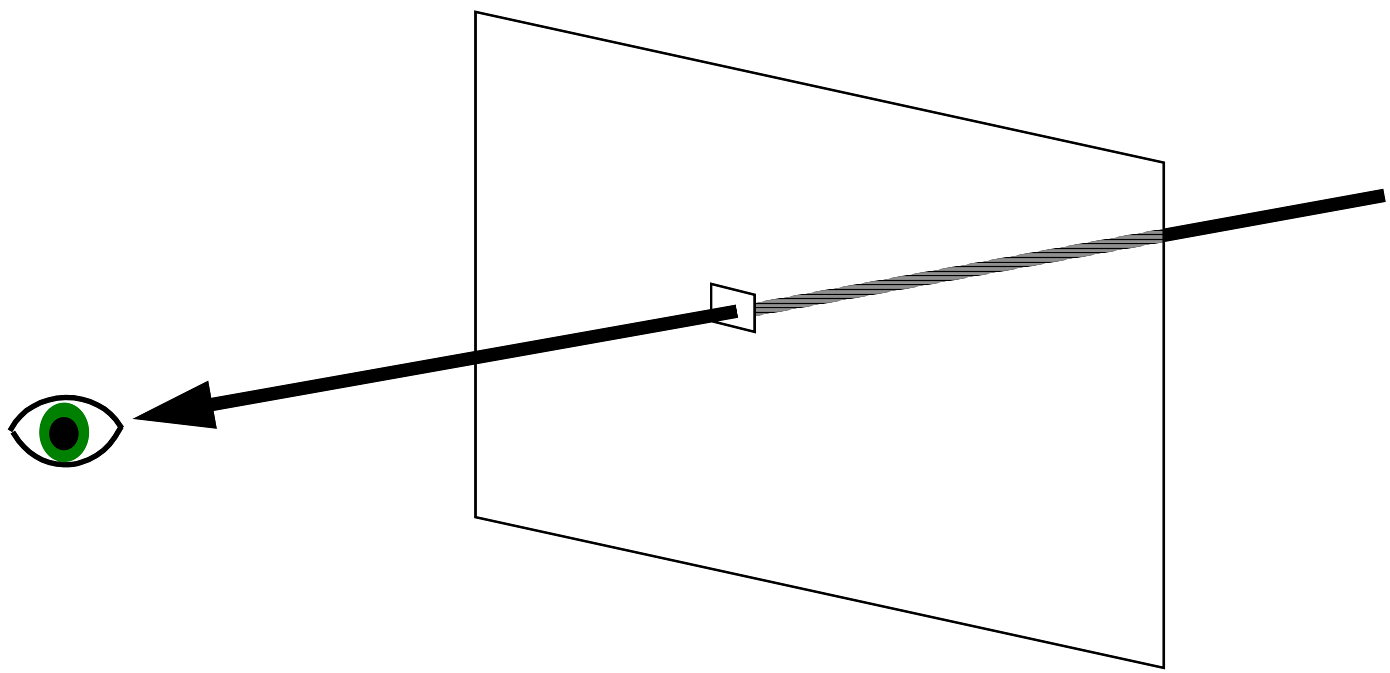
Radiometric Quantities

- **Irradiance** E
  - **Quantity of light incident to a surface point**
  - **Incident radiant flux / area (Watt / meter^2)**
  - **Radiance integrated over hemisphere**



From Radiance to Pixels

- Compute radiance incident to eye through each screen pixel

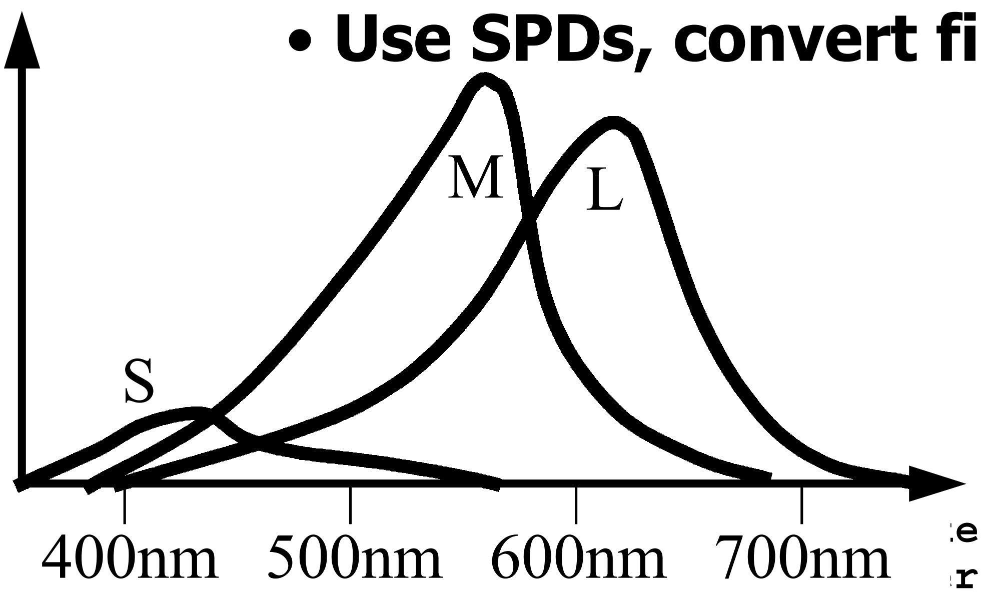


From Radiance to Pixels

- **Pixel value based on radiance**
- **But radiance is distributed continuously along the spectrum**
  - **We need three numbers: R, G, B**

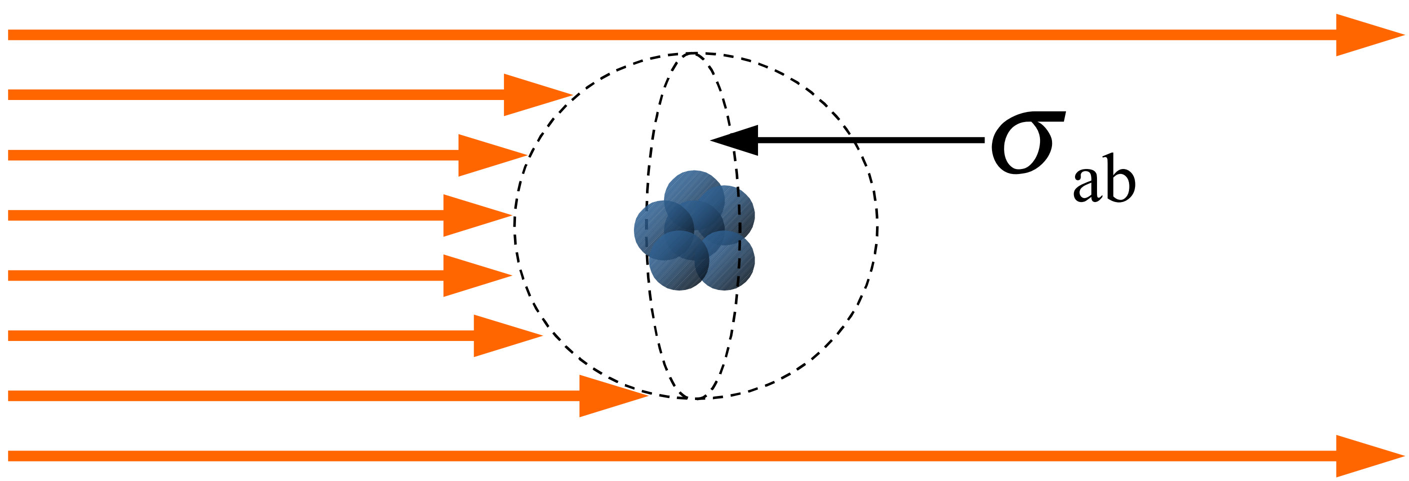
From Radiance to Pixels

- **SPD (Spectral Power Distribution) to RGB**
  - **Fast approach:**
    - **Do all math at R, G, B sample wavelengths**
  - **Correct approach:**
    - **Use SPDs, convert final radiance to RGB**



Absorption

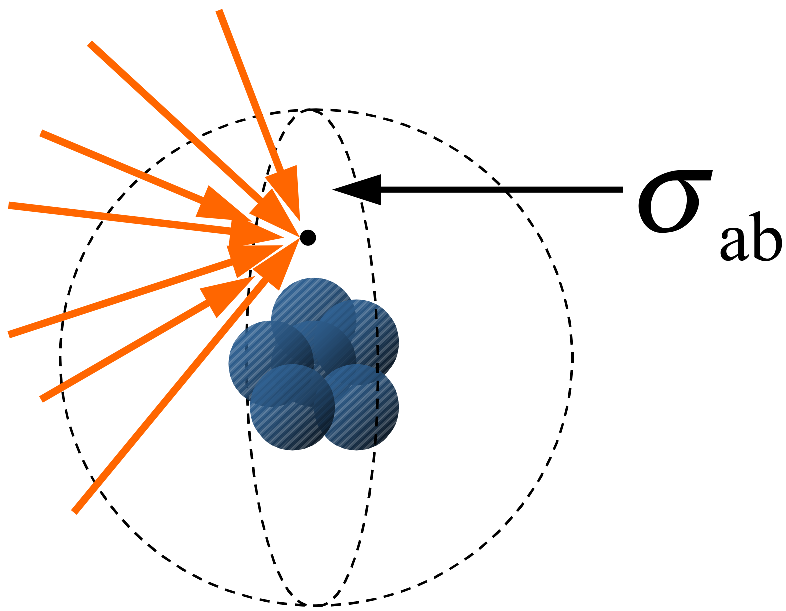
- Absorption cross section σ_{ab}
  - Absorbed radiant flux per unit incident irradiance Φ / E
  - Units of area (meter²)



Absorption

- **Absorption cross section** σ_{ab}

$$\Phi = E\sigma_{ab} \Longrightarrow \sigma_{ab} = \Phi / E$$



Absorption

- Absorption coefficient β_{ab}
  - Particle density ρ_{ab} times absorption cross section σ_{ab}
  - Units of inverse length (meter^{-1})

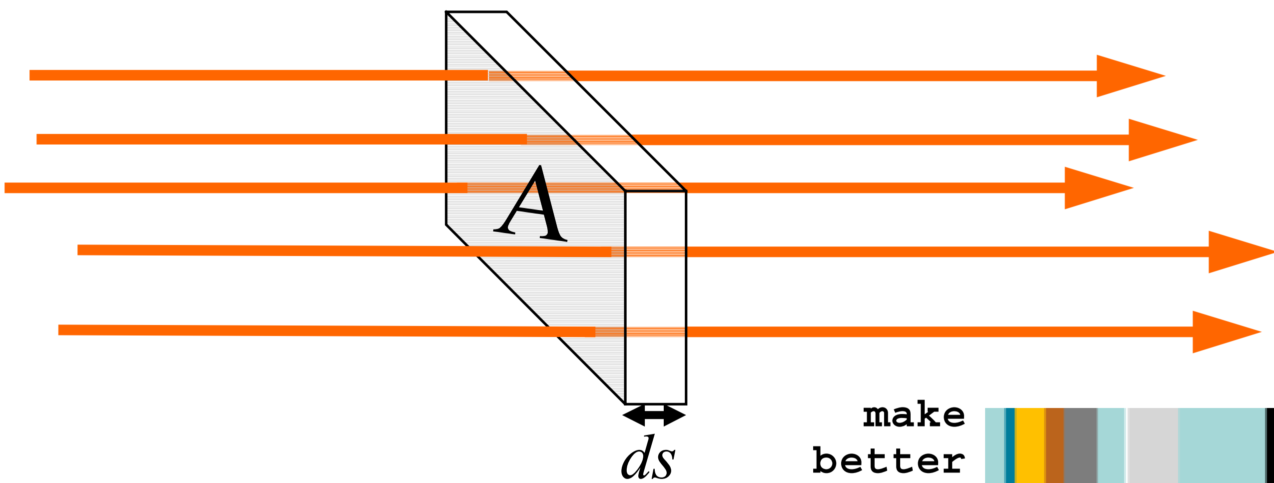
Absorption

- **Total absorption cross section:**

$$A_{ab} = \sigma_{ab} \rho_{ab} A \, ds$$

- **Probability of absorption:**

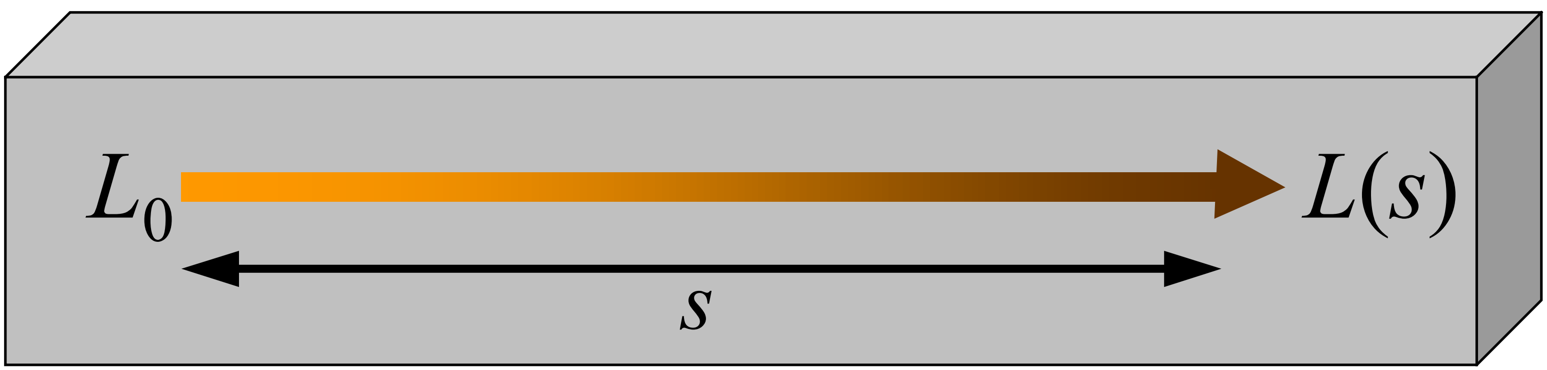
$$P_{ab} = A_{ab} / A = \sigma_{ab} \rho_{ab} \, ds = \beta_{ab}$$



Absorption

- **Attenuation of radiance from travel through a <u>constant-density</u> absorptive medium:**

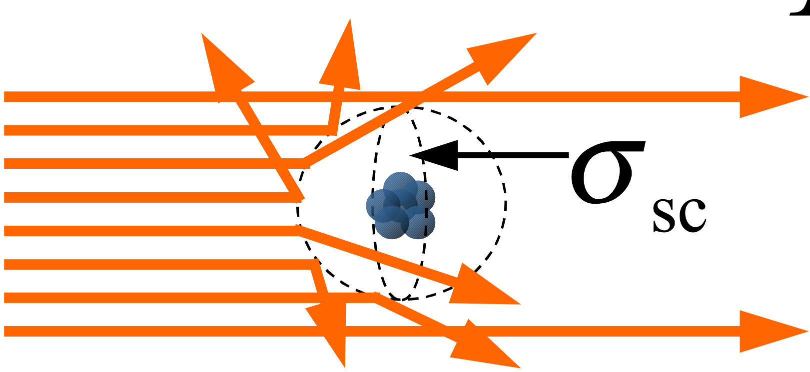
$$L(s) = L_0 e^{-\beta_{ab} s}$$



Out-Scattering

- Exactly as in the absorption case
  - Scattering cross section σ_{sc}
  - Scattering coefficient $\beta_{sc} = \rho_{sc}\sigma_{sc}$
  - Attenuation due to out-scattering in a constant-density medium:

$$L(s) = L_0 e^{-\beta_{sc} s}$$



Extinction

- **Both absorption and out-scattering attenuate light**
- **They can be combined as extinction**
- **Extinction coefficient** $\beta_{\mathrm{ex}} = \beta_{\mathrm{ab}} + \beta_{\mathrm{sc}}$
- **Total attenuation from extinction**

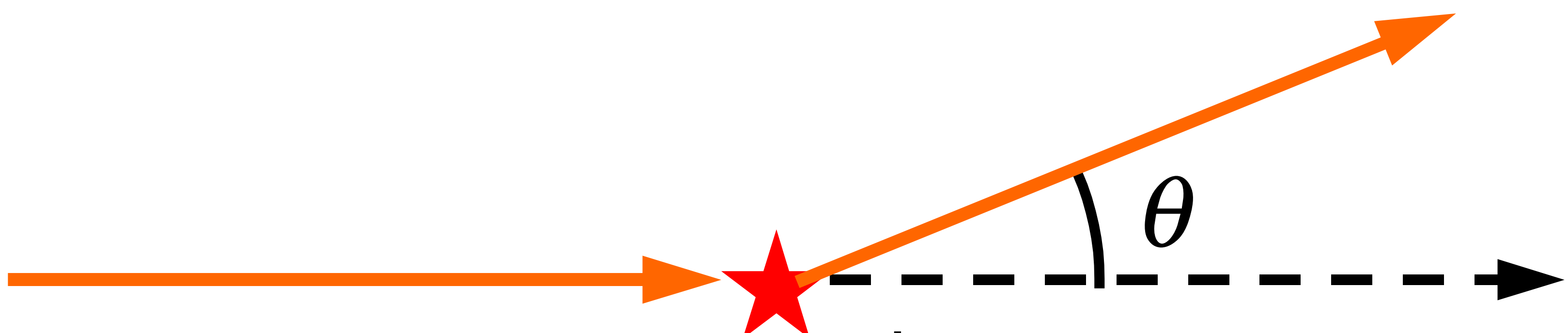
$$L(s) = L_0 e^{-\beta_{\mathrm{ex}} s} \Longrightarrow F_{\mathrm{ex}}(s) = e^{-\beta_{\mathrm{ex}} s}$$

In-Scattering

- Light is scattered into a view ray from all directions
  - From the sun
  - From the sky
  - From the ground
- We will only handle in-scattering from the sun

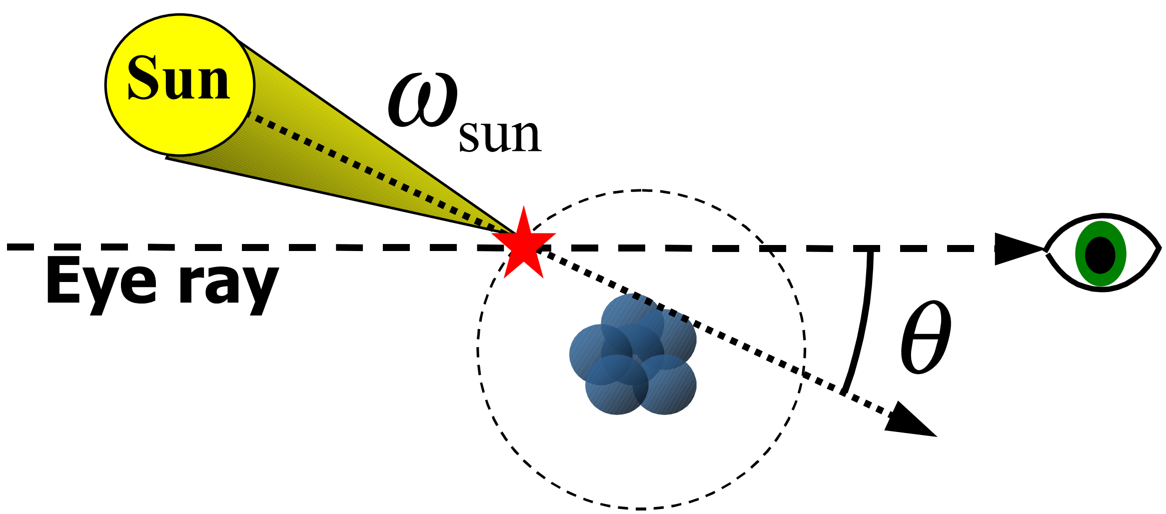
In-Scattering

- **Where does a scattered photon go?**
  - **Scattering phase function** $f(\theta, \varphi)$
    - **If a photon is scattered, gives the probability it goes in direction** θ, φ
    - **Most atmospheric particles are spherical or very small:** $f(\theta, \varphi) = f(\theta)$



In-Scattering

- How do we use $f(\theta)$ for in-scattering?
  - In-scatter probability: $f(\theta)\omega_{\text{sun}}$
  - In-scatter radiance : $f(\theta)\omega_{\text{sun}} L_{\text{sun}} = f(\theta) E_{\text{sun}}$



In-Scattering

- **In-scattering over a path**
  - **Radiance from a single event:** $f(\theta)E_{\mathrm{sun}}$
  - **Over a distance** ds**:** $f(\theta)E_{\mathrm{sun}}\beta_{\mathrm{sc}}ds$
- **Angular scattering coefficient**

$$\beta_{\mathrm{sc}}(\theta) = \beta_{\mathrm{sc}} f(\theta)$$

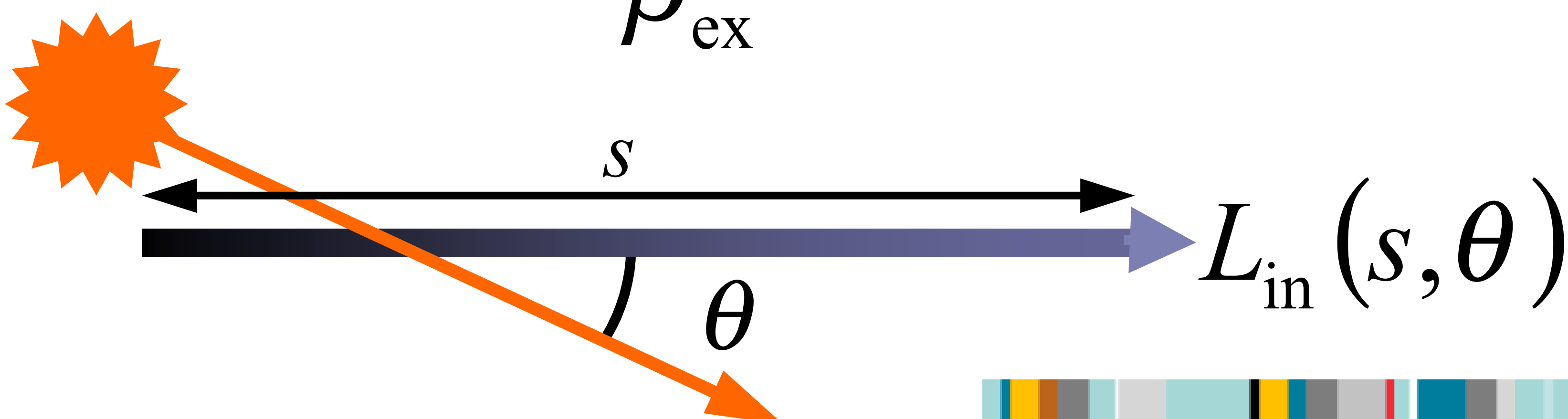
  - **In-scattering over** ds**:** $E_{\mathrm{sun}}\beta_{\mathrm{sc}}(\theta)ds$
  - **Units of** $\beta_{\mathrm{sc}}(\theta)$**: meter^{-1} * steradian^{-1}**

GameDevelopers Conference

In-Scattering

- **Added radiance from solar in-scattering through a constant-density scattering medium:**

$$L_{\mathrm{in}}(s,\theta) = \frac{1}{\beta_{\mathrm{ex}}} E_{\mathrm{sun}} \beta_{\mathrm{sc}}(\theta)\left(1 - e^{-\beta_{\mathrm{ex}} s}\right)$$

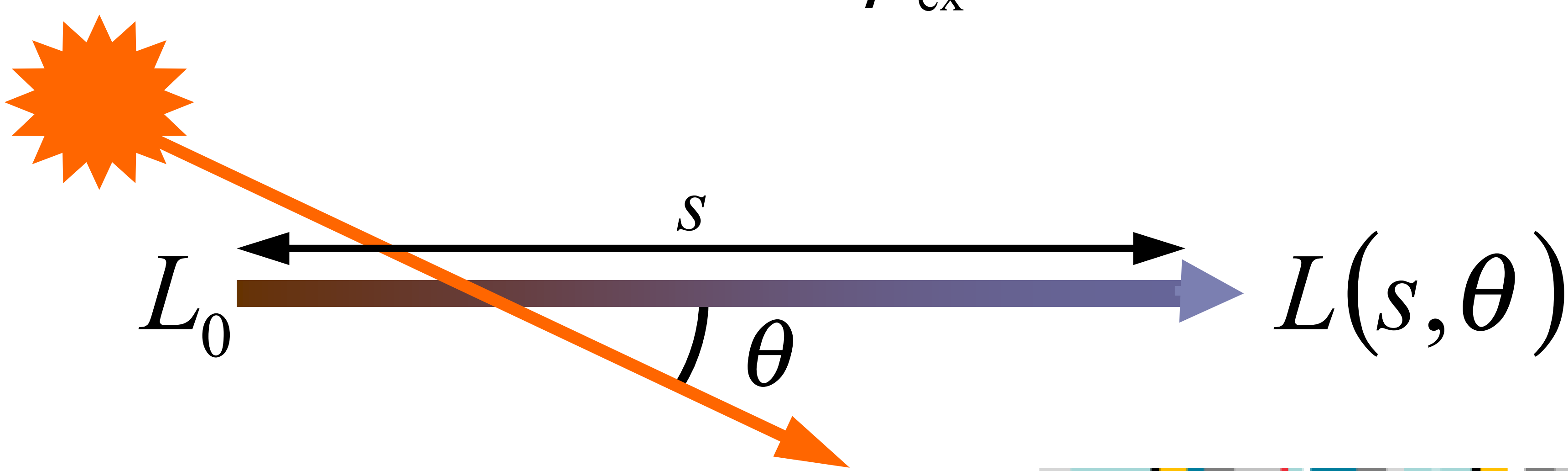


games

Extinction and In-Scattering

$$L(s,\theta) = L_0 \mathrm{F}_{\mathrm{ex}}(s) + L_{\mathrm{in}}(s,\theta)$$

$$\mathrm{F}_{\mathrm{ex}}(s) = e^{-\beta_{\mathrm{ex}} s} \qquad L_{\mathrm{in}}(s,\theta) = \frac{1}{\beta_{\mathrm{ex}}} E_{\mathrm{sun}} \beta_{\mathrm{sc}}(\theta)\left(1 - e^{-\beta_{\mathrm{ex}} s}\right)$$



Extinction and In-Scattering

$$L(s,\theta) = L_0 \mathrm{F}_{\mathrm{ex}}(s) + L_{\mathrm{in}}(s,\theta)$$

- **Compare to hardware fog:**

$$L(s) = L_0(1 - \mathrm{f}(s)) + C_{\mathrm{fog}}\mathrm{f}(s)$$

  - **Monochrome extinction**
  - **Added color completely non-directional**

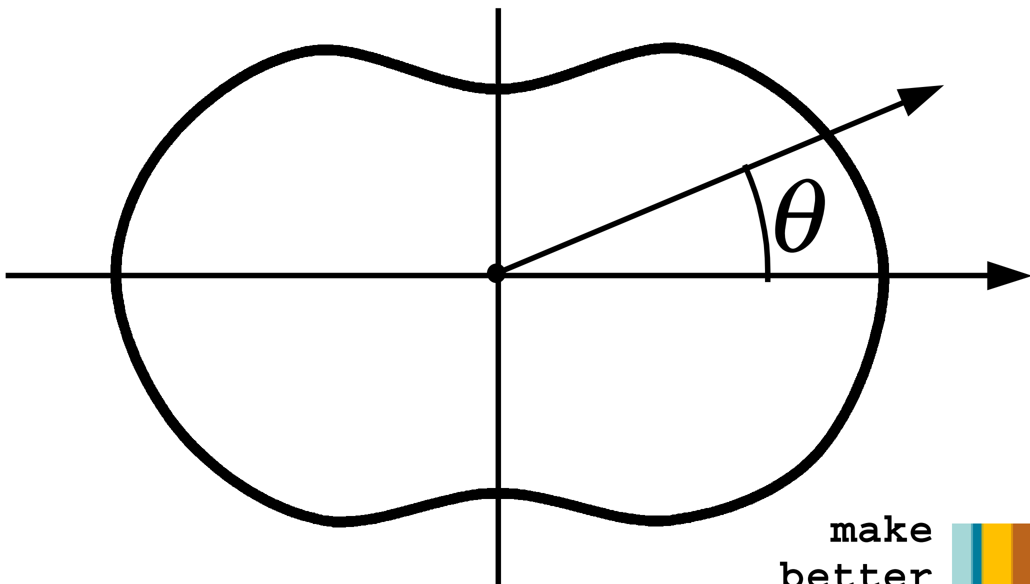
Rayleigh Scattering

- **Small particles** $\left(r < 0.05\lambda\right)$
- β_{sc} **is proportional to** λ^{-4}

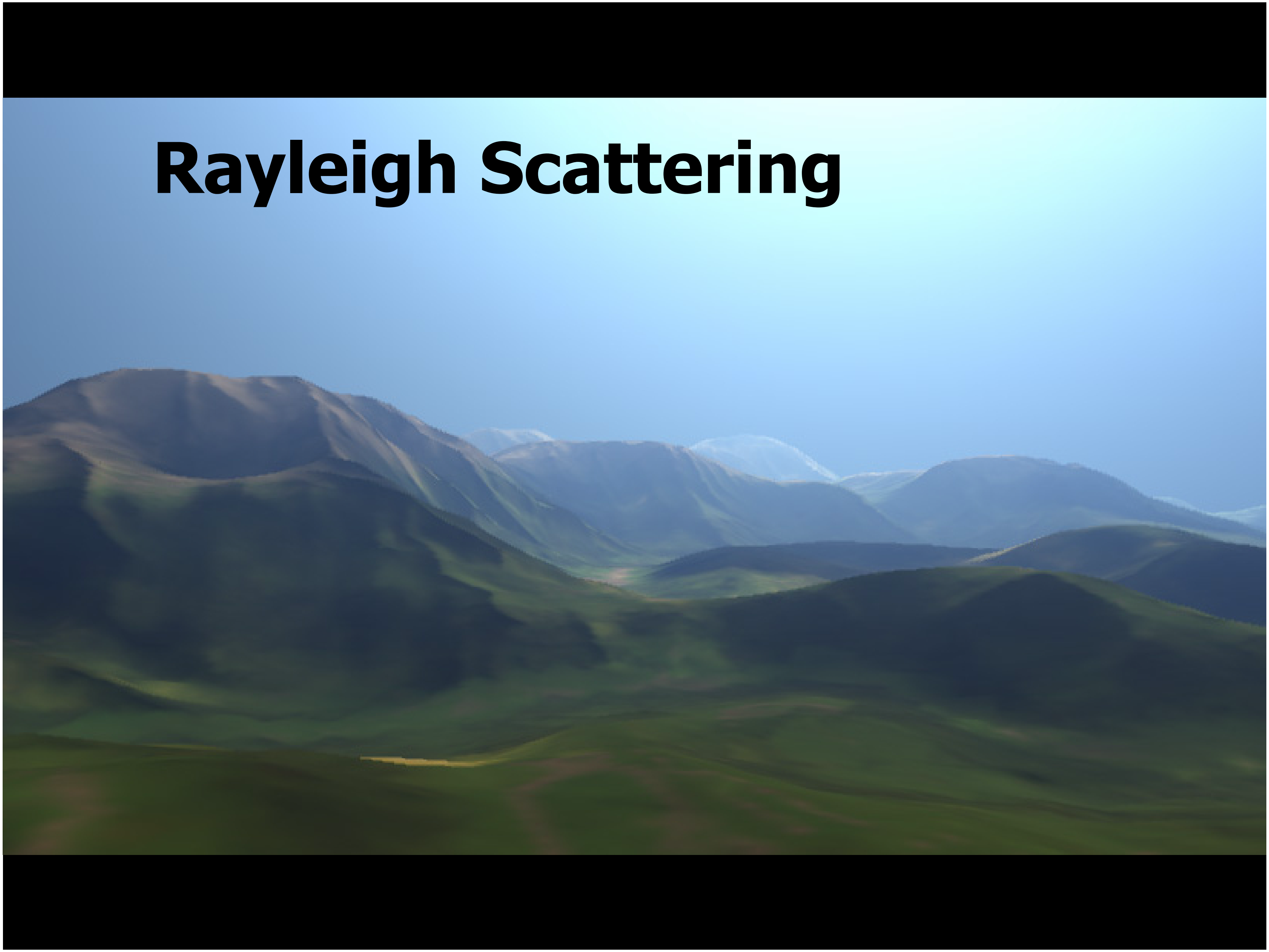
Rayleigh Scattering

- **Phase function:**

$$f_R(\theta) = \frac{3}{16\pi}\left(1 + \cos^2\theta\right)$$



Rayleigh Scattering

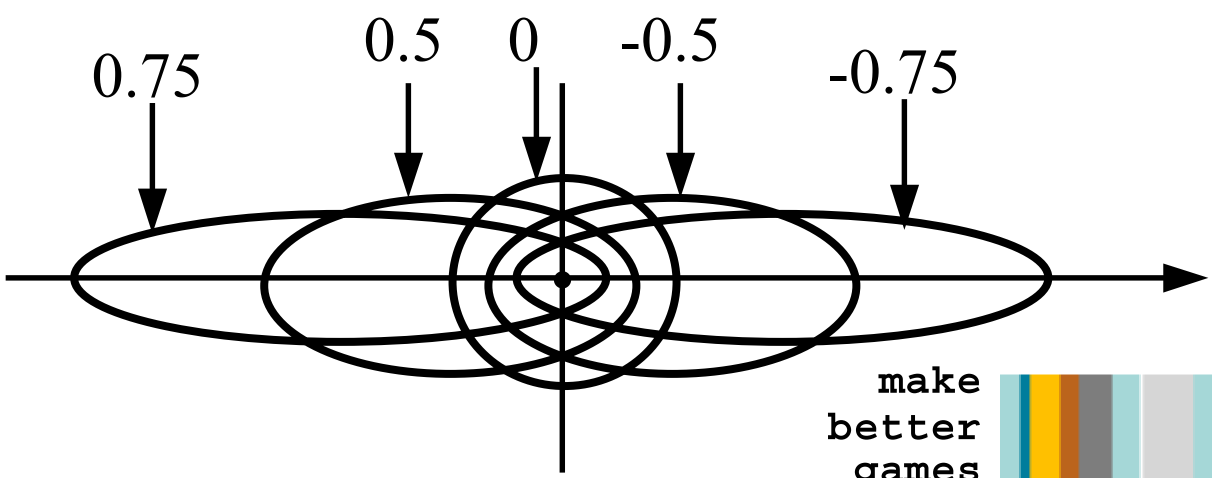


Game Developers Conference

Mie Scattering

- **Larger, spherical particles**
- **Phase function approximation:**
  - **Henyey-Greenstein**

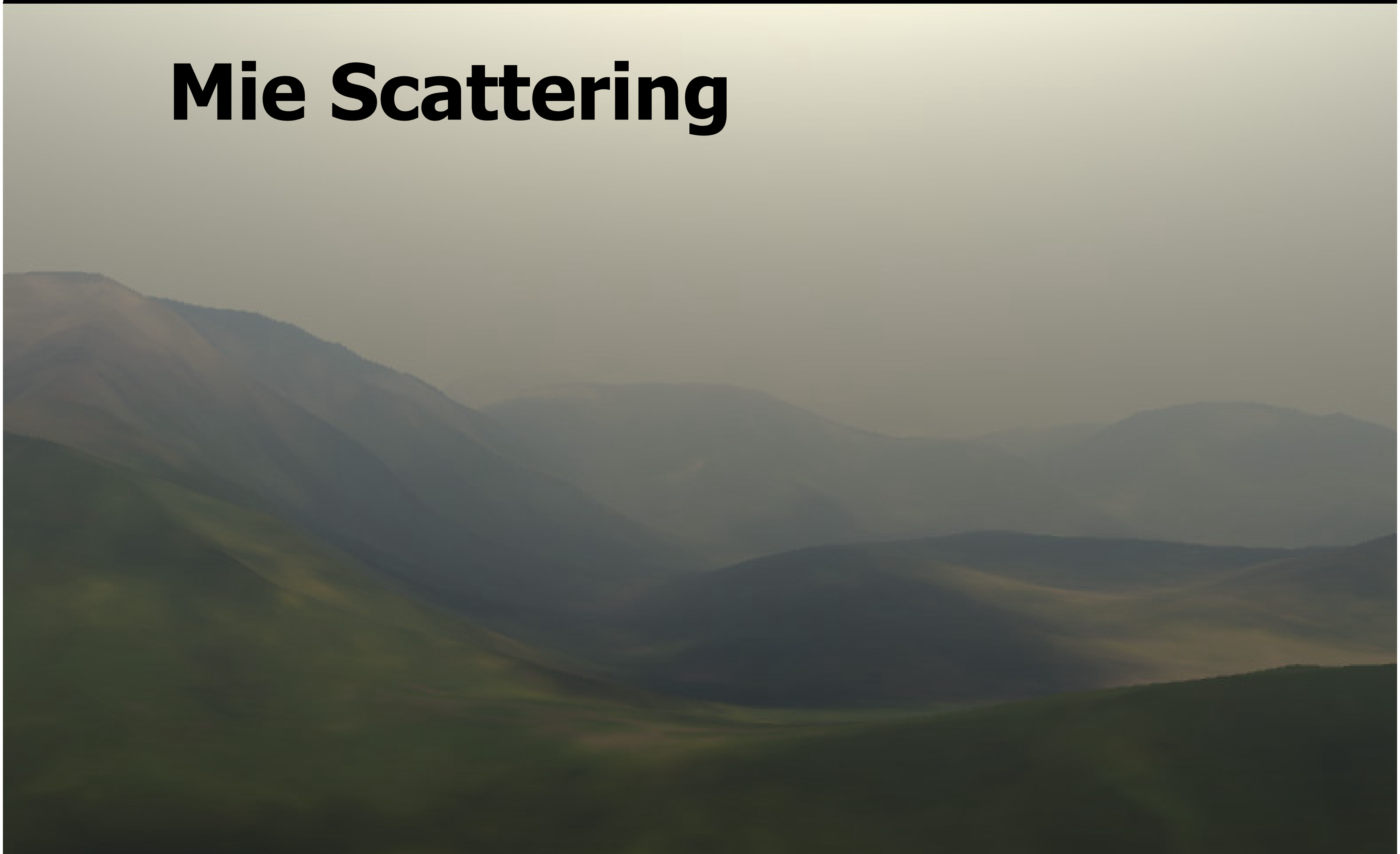
$$f_{\mathrm{HG}}(\theta) = \frac{(1-g)^2}{4\pi\left(1+g^2-2g\cos(\theta)\right)^{3/2}}$$



Mie Scattering

- Wavelength dependence
  - Complex and depends on exact size of particle
  - In practice, air usually contains a mix of various sizes of Mie particles
    - In the aggregate these tend to average out any wavelength dependence

Mie Scattering



Combined Scattering

- In practice, air contains both Rayleigh and Mie scatterers
- Absorption is usually slight
- We will use:

$$\beta_{\text{ex}} = \beta_{\text{sc}}^{\text{Rayleigh}} + \beta_{\text{sc}}^{\text{Mie}}$$

$$\beta_{\text{sc}}(\theta) = \beta_{\text{sc}}^{\text{Rayleigh}} f_{\text{R}}(\theta) + \beta_{\text{sc}}^{\text{Mie}} f_{\text{HG}}(\theta)$$

Parameters

- **Atmospheric parameters:**

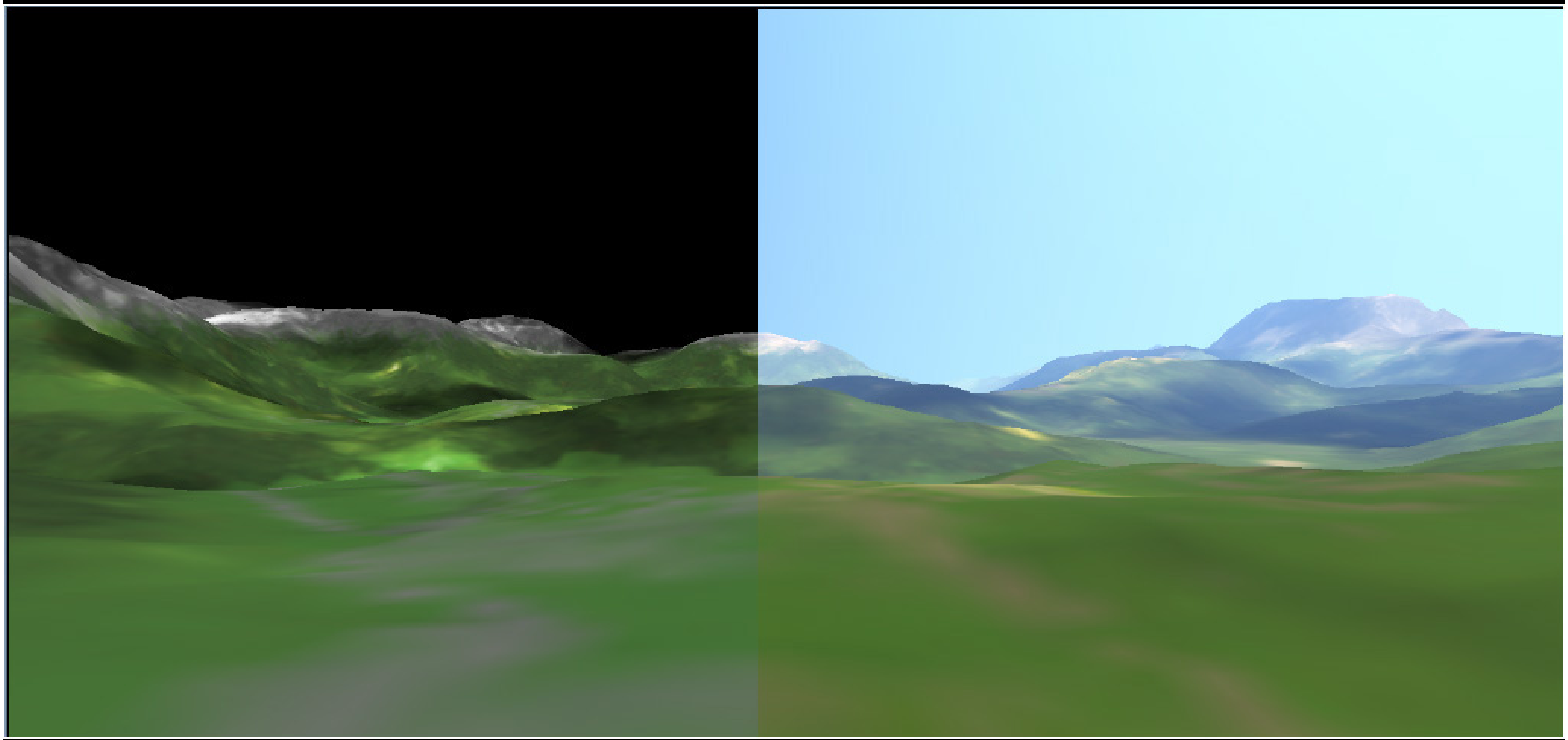
$$\beta_{sc}^{\text{Rayleigh}} \qquad \beta_{sc}^{\text{Mie}} \qquad g_{\text{HG}}$$

- **Constant?** E_{sun}
  - **Affected by extinction**

- **Constant:** E_{sun}^{0}

Implementation

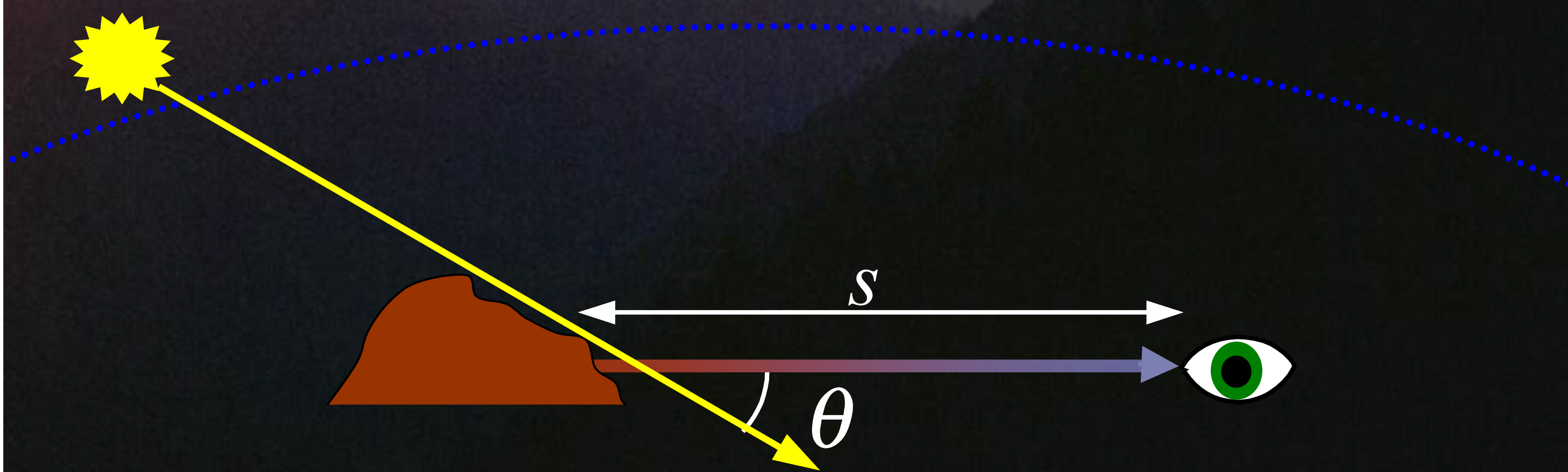
How ?



Without scattering   With scattering

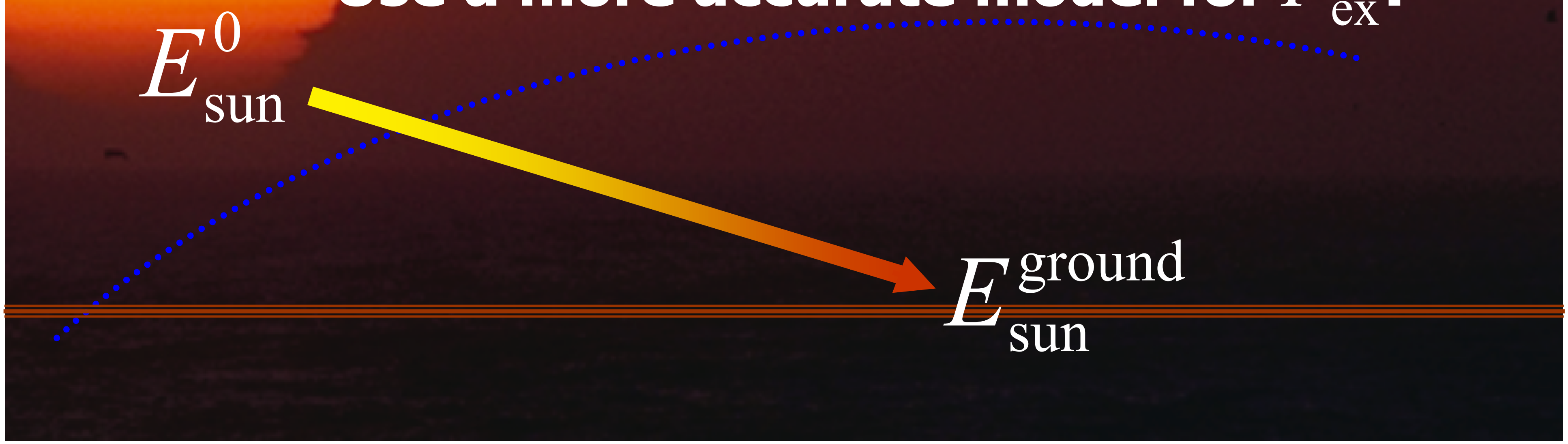
Implementation

- Aerial Perspective
  - Extinction & Inscattering
  - Rays low in atmosphere
  - Constant density good approximation



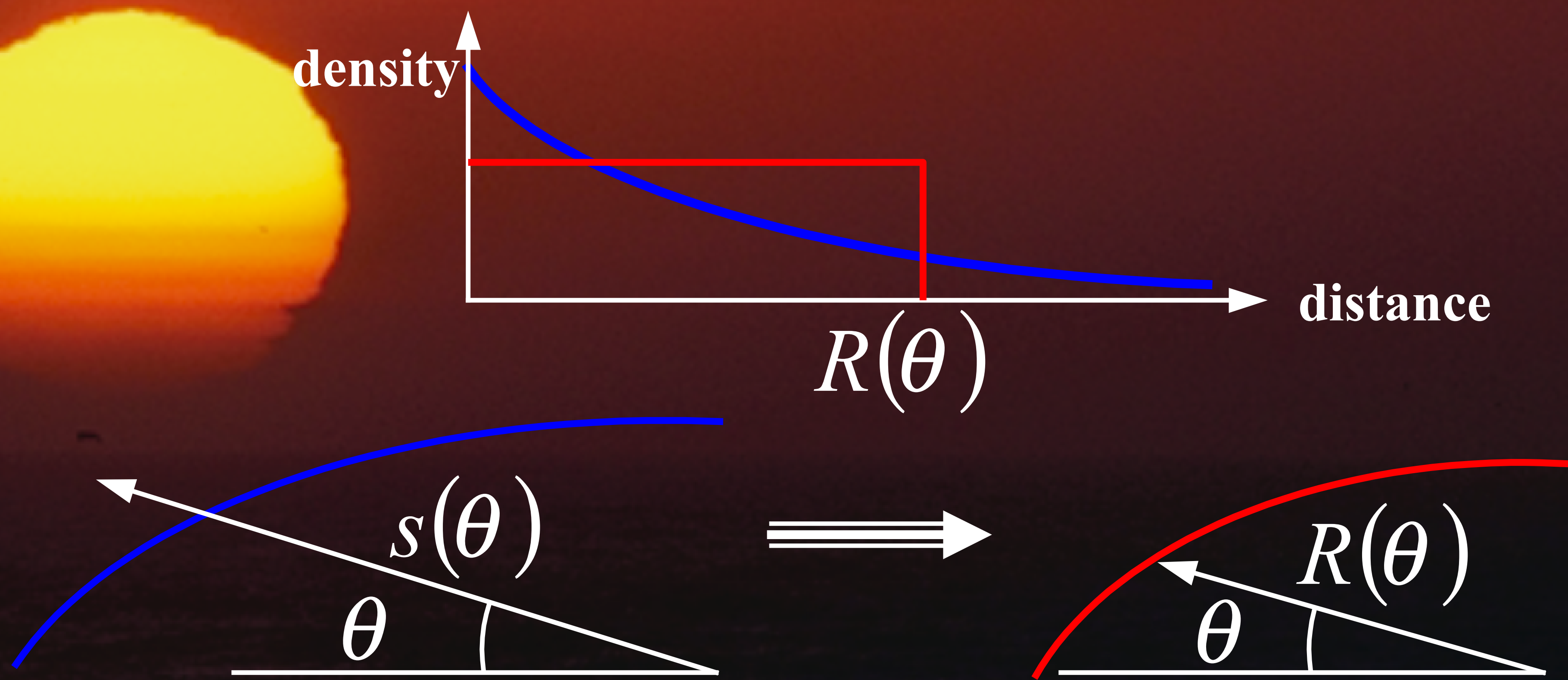
Implementation

- **Sunlight** $E_{\text{sun}}^{\text{ground}} = E_{\text{sun}}^{0} F_{\text{ex}}$
  - E_{sun}^{0} **is white**
  - **Density is <u>not</u> constant!**
  - **Use a more accurate model for F_{ex}?**



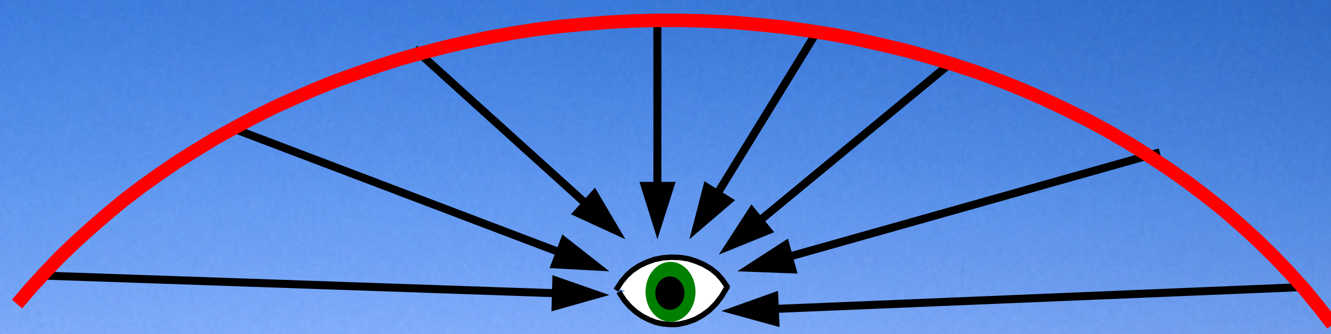
Implementation

- **Sunlight:**
  - **Virtual sky dome, use simple model**



Implementation

- **Sky color:** $L_{\mathrm{sky}}(\theta,\varphi)=F_{\mathrm{in}}(\theta,\varphi)$
  - **Density is <u>not</u> constant!**
  - **More accurate model too expensive**
    - **Many computations needed per frame**
- **Sky geometry**
  - **Virtual sky dome**



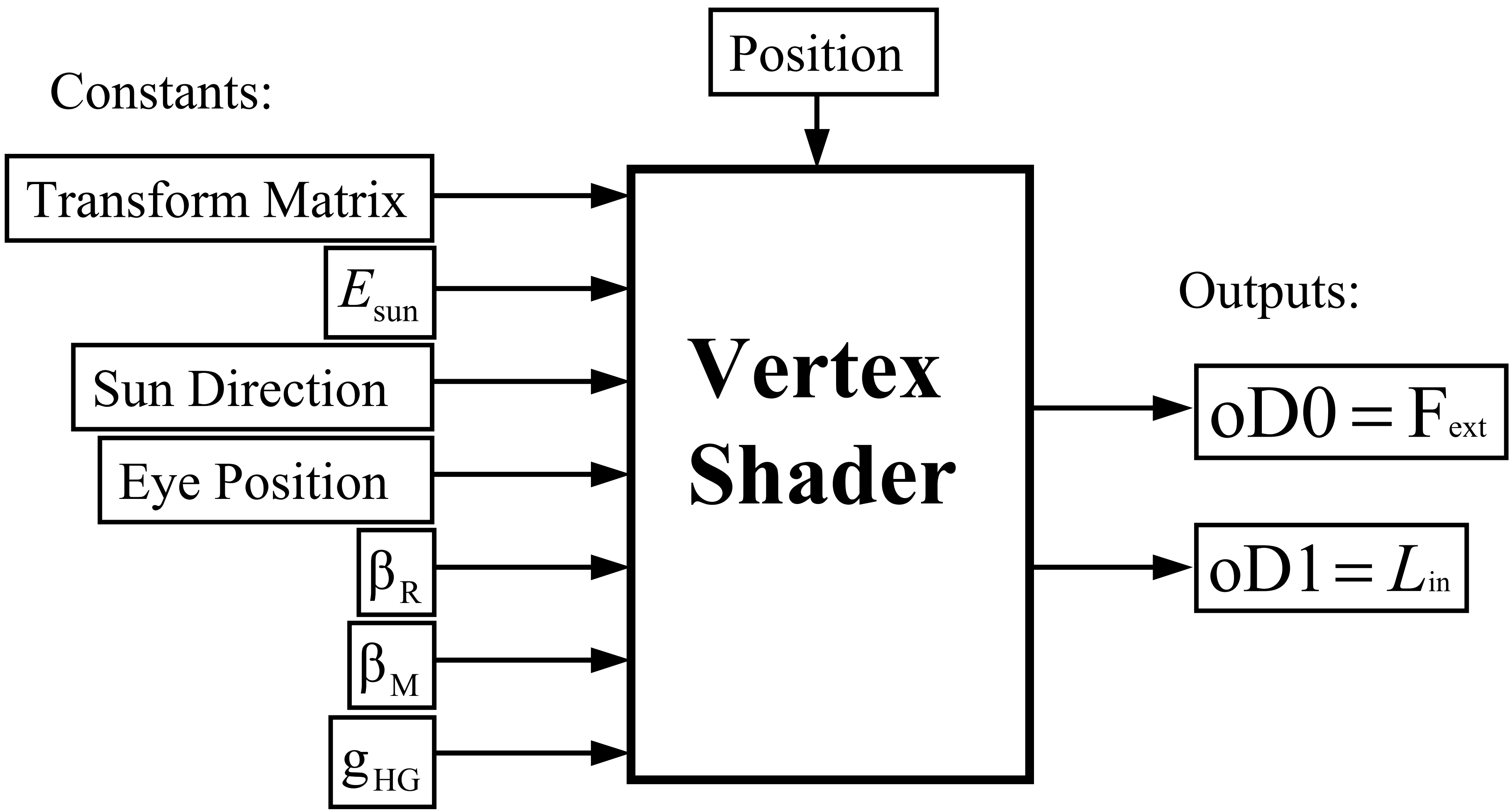
Implementation

- **Compute:** $L(s,\theta) = L_0 \mathrm{F}_{\mathrm{ex}}(s) + L_{\mathrm{in}}(s,\theta)$
  - **Can be done with textures**
    - **1D texture for** F_{ex}
      - **Texture coordinate is a function of** s
    - **2D texture for** L_{in}
      - **Texture coords are functions of** s, θ
    - **Combine in pixel shader**
  - **We decided on a different approach**

Implementation

- **Compute:** $L(s,\theta) = L_0 F_{ex}(s) + L_{in}(s,\theta)$
  - **Use vertex shader to compute** F_{ex}, L_{in}
  - **Apply as vertex interpolated colors**
    - **In pixel shader, or even fixed pipeline**
  - **Pros:**
    - **Doesn't use valuable texture slots**
    - **Can change atmosphere properties**
  - **Cons:**
    - **Somewhat dependent on tessellation**

Vertex Shader



Vertex Shader

$$L(s,\theta) = L_0 \mathrm{F}_{\mathrm{ex}}(s) + L_{\mathrm{in}}(s,\theta)$$

$$\mathrm{F}_{\mathrm{ex}}(s) = e^{-(\beta_{\mathrm{R}} + \beta_{\mathrm{M}})s}$$

$$L_{\mathrm{in}}(s,\theta) = \frac{\beta_{\mathrm{R}}(\theta) + \beta_{\mathrm{M}}(\theta)}{\beta_{\mathrm{R}} + \beta_{\mathrm{M}}} E_{\mathrm{sun}} \left(1 - e^{-(\beta_{\mathrm{R}} + \beta_{\mathrm{M}})s}\right)$$

$$\beta_{\mathrm{R}}(\theta) = \frac{3}{16\pi} \beta_{\mathrm{R}} \left(1 + \cos^2 \theta\right)$$

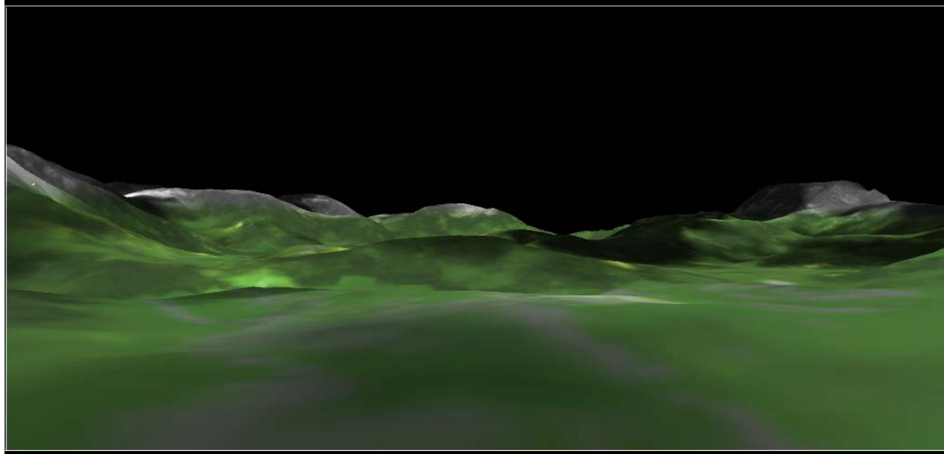
$$\beta_{\mathrm{M}}(\theta) = \frac{1}{4\pi} \beta_{\mathrm{M}} \frac{(1-g)^2}{\left(1 + g^2 - 2g\cos(\theta)\right)^{3/2}}$$

Vertex Shader

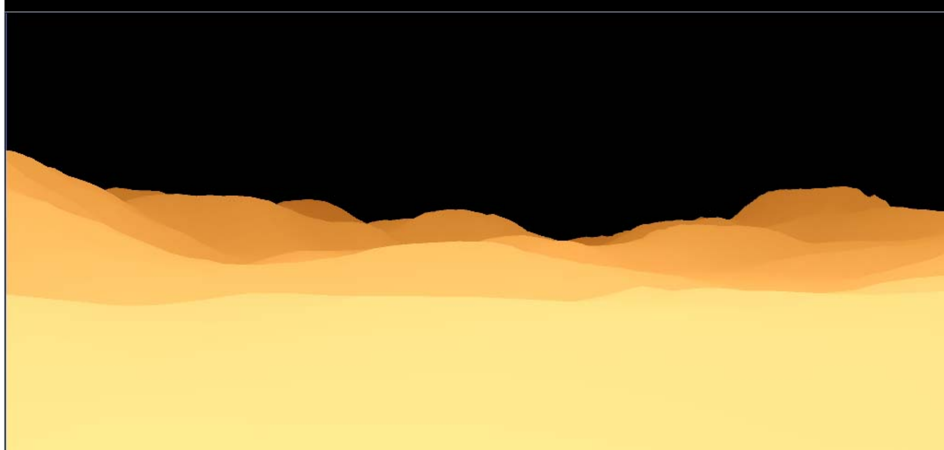
- Current Implementation:
  - 33 Instructions
    - Not including macro expansion
    - Could probably be optimized
  - 8 registers

Pixel Shader

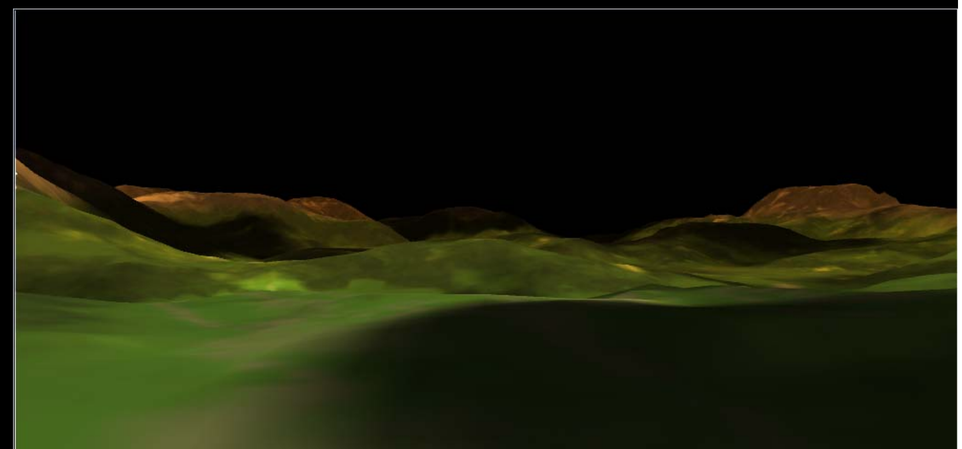
$$L = L_0 * F_{ex} + L_{in}$$



L_0 x =



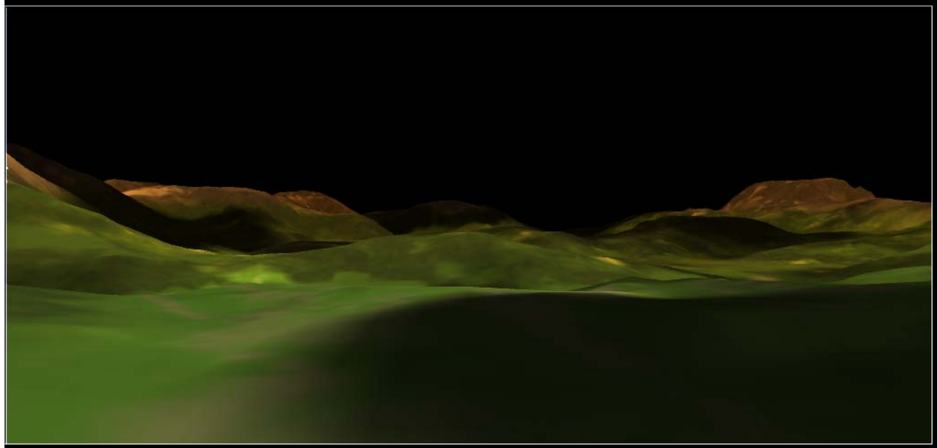
F_{ex}



$L_0 * F_{ex}$

Pixel Shader

$$L = L_0 * \mathrm{F}_{ex} + L_{in}$$



$L_0 * \mathrm{F}_{ex}$

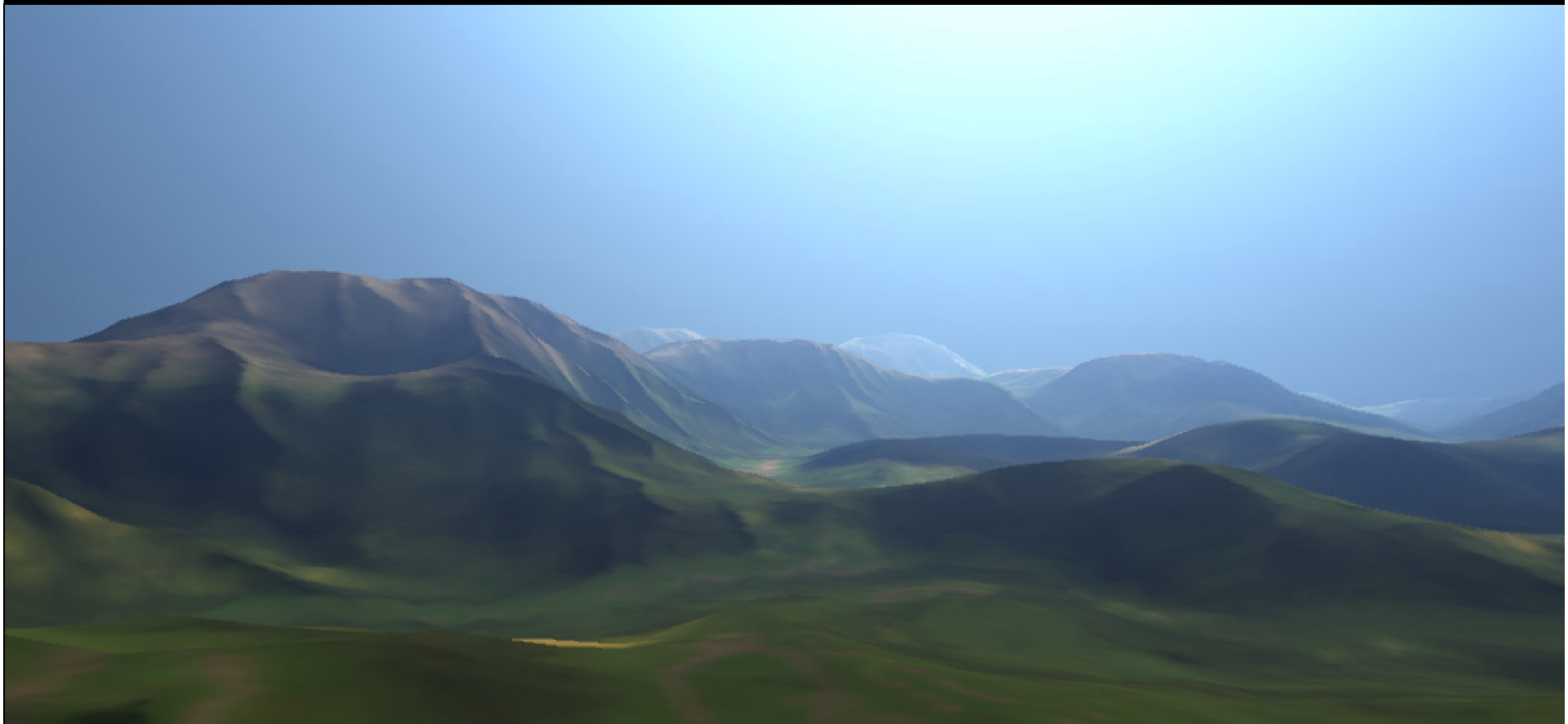
+

=

$L = L_0 * \mathrm{F}_{ex} + L_{in}$

L_{in}

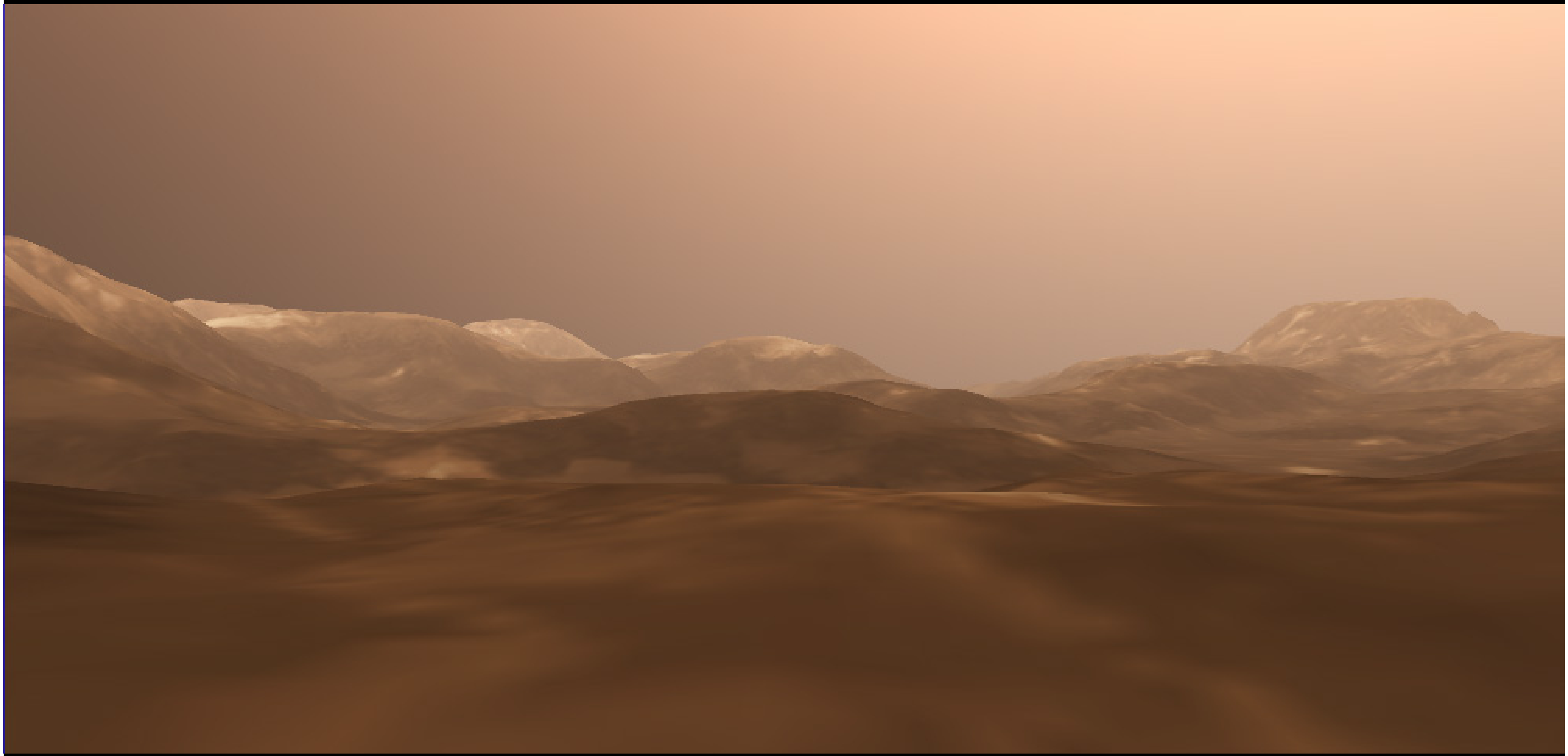
Results



Rayleigh Scattering - high
Mie Scattering - low
Sun Altitude - high

Rayleigh Scattering - low
Mie Scattering - high
Sun Altitude - high

Rayleigh Scattering - medium
Mie Scattering - medium
Sun Altitude - low



Planet Mars like scattering

Demo

Conclusion

- **Scattering is easy to implement.**
- **Easy to add to an existing rendering framework**
  - **compute** F_{ex} **and** L_{in}
  - **apply these to existing color to get final color**

Future Work

- In-scattering from sky
- Clouds (scattering and extinction)
- Volumetric cloud shadows
- Non-uniform density distributions
- Full-spectrum math?

Acknowledgements

- We would like to thank
  - Kenny Mitchell for the terrain engine used in our demo
  - Solomon Srinivasan for help with the demo movie

References

[Blinn1982] J. F. Blinn. *Light Reflection Functions for Simulation of Clouds and Dusty Surfaces.*

[Dutr 2001] P. Dutr . *Global Illumination Compendium.*

[Henyey1941] L. G. Henyey and J. L. Greenstein. *Diffuse Reflection in the Gala y.*

[Hoffman2001] N. Hoffman and K. J. Mitchell. *Photorealistic   errain Lighting in Real   ime.*

[Klassen1987] R. V. Klassen. *Modeling the Effect of the Atmosphere on Light.*

[Mie1908] G. Mie. *Bietage zur Optik truber Medien Speziell Kolloidaler Metallosungen.*

[Preetham1999] A. J. Preetham, P.   hirley, B.   .   mits. *A Practical Analytic Model for Daylight.*

[Rayleigh1871] J. W.   trutt Lord Rayleigh . *On the light from the sky, its polarization and colour.*

[Yee2002] H. Yee, P. Dutr ,   . Pattanai . *Fundamentals of Lighting and Perception   he Rendering of Physically Accurate Images*.

THANK YOU