Lunar Psychophysics: Rendering Planetary Atmospheres in Virtual Reality

THE CATHOLIC UNIVERSITY OF AMERICA | Department of Psychology

Lunar Psychophysics Virtual Reality Laboratory

Department of Psychology, The Catholic University of America, Washington, D.C.

Matt Incao, Francesca Gualano, Rebecca Johnson, Mary Funk, Patrick Miller, Katherine Rahill M.A. & Marc Sebrechts, Ph.D.

THE CATHOLIC UNIVERSITY OF AMERICA | Department of Psychology

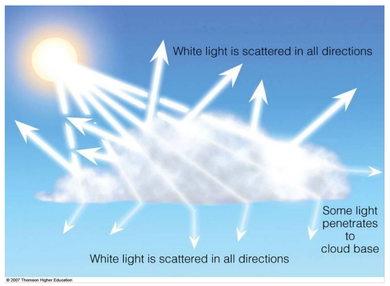
Introduction

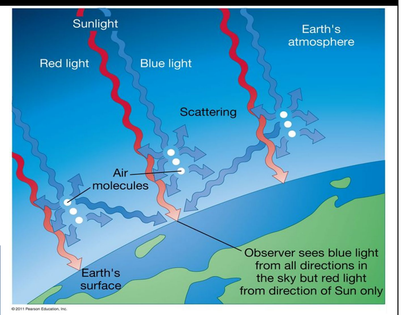
Generating an accurate visual representation of atmospheric scattering in computer graphics models (CGM) is challenging, and understanding these equations is crucial for rendering realistic Virtual Environments (VEs). Since the equations used explain atmospheric light scattering are extremely complex, CGM models generally use simplified equations. Fortunately, we can numerically compute scattering capabilities , as suggested by O'Neil (2005), who proposed the implementation of full scattering equations to model an atmosphere more accurately in Virtual Reality (VR).

This poster presents the ongoing work in the Lunar Psychophysics Virtual Reality (LPVR) Lab on novel approaches for atmospheric rendering in VR. Specifically, it provides a theoretical review on the process of simulating surface textures and the scattering mechanisms to create a planetary-like atmosphere such as the Moon or Mars.

Rayleigh and Mie Scattering

Earth's atmosphere is constructed of unique properties that consist of a specific range of uniform size particles. The scattering of light occurs when light strikes a range of small, non-absorbing spherical particles, known as ***Rayleigh scattering***, (Figure 1, right) which results in the omnidirectional scattering of light across the surface and is responsible for the blue sky and the Sun's yellow hue.



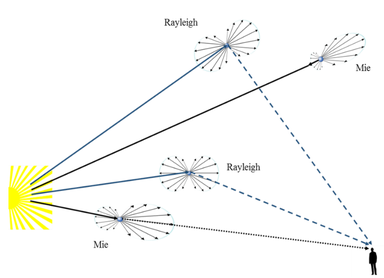


Unlike Rayleigh scattering, ***Mie scattering*** (Figure 2, left) assumes particles of much larger sizes, creating scatter properties which are not uniformly distributed. This is due to the complex interactions in result of refracted rays, particle size and varying reflectance properties. In instances when all wavelengths of light are scattered equally, Mie scattering is occurring; hence why clouds appear white.

Observational Light and The Observer

A goal of lunar psychophysics is to apply the functions of atmospheric scattering to understand perceptual distortions by referencing the location of the light source and an object relative to the observer. This is because visibility can change based on the intensity and direction of light relative to the observer and create inconsistencies in the visibility of an object's surface features.

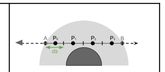
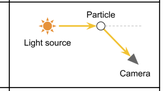
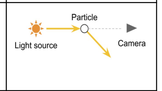
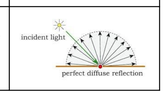
Figure 3 (below) is a conceptual illustration showing different hypothetical objects in the sky with different reflectance properties: Rayleigh (small particulate matter) and Mie (larger particulate matter). The two Rayleigh objects (blue dotted lines) objects illustrate a uniform distribution of reflectance that reaches the observer in a consistent, perceivable wavelength (blue)



The two Mie objects (black lines) illustrate two different examples of the characteristic inconsistencies of Mie scatter that vary based on an object's location relative to the observer. The triangulated position of an object's relative location to the Sun and the observer can result in the forward scattering of light off an object, causing backscatter (bottom left). Or, it can result in the distributed illuminance of the second object (top right) causing light to refract in a different direction that does not reach the observer; concealing the object almost completely (Nave, 2005).

Atmospheric Scattering in Unity 3D

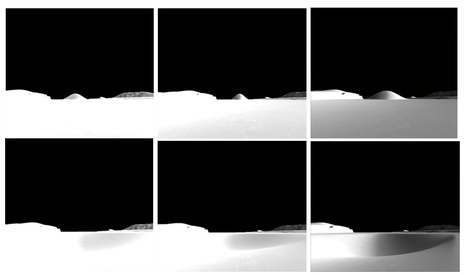
To create a realistic, accurate representation of the visual effects of light scattering on Earth and the Moon, each of these VR models are further modified as needed by adjusting the following variables in Unity 3D: Rayleigh scatter/extinction coefficients, Mie Scatter/extinction coefficients, shaders, atmosphere height and density, sun elevation and intensity, density scale and height of terrain. For the use of rendering an atmospheric scattering model the Heney-Greenstein Phase Function, followed by the phase function for Elek should be used to find the atmospheric density. Integration in-scattering should then be performed as well as computing optical depth. The 4 main equations used to simulate atmospheric light scattering are below

Adaptation of the Heney-Greenstein Phase Function $F(\theta, g) = \frac{3 \times (1 - g^2)}{2 \times (2 + g^2)} \times \frac{1 + \cos^2 \theta}{(1 + g^2 - 2 \times g \times \cos \theta)^{\frac{3}{2}}}$	Light is equal to the amount of light emitted from the sun, multiplied by the sum of the individual contributions from each point in the segment	
Out-Scattering $t(P_a P_b, \lambda) = 4\pi \times K(\lambda) \times \int_{P_a}^{P_b} exp\left[\frac{-h}{H_0}\right] ds$	Occurs when a ray of light that was directed towards the camera is deflected away from it	
In-scattering $I'_v() = I_y() + I_e() \times exp(-t(P_a P_b,))$	Occurs when a ray of light is deflected directly towards the camera.	
Surface Scattering $I_P(\lambda) = I_I(\lambda) \times K(\lambda) \times F(\theta, g) \times \int_{P_a}^{P_b} \left[exp\left[\frac{-h}{H_0}\right] \times exp(-t(PP_c, \lambda) - t(PP_a, \lambda))\right] ds$	Accounts for the color of out-scattering light that takes place before the sunlight strikes the surface	

Rendering a Lunar Atmosphere

The characterization of the scattering of sunlight on the Moon has been modeled using Mie scatter models. However, when Mie scattering coefficients were adjusted to "best fit" the visual representation of a lunar VE in Unity 3D, the visual effects did not match lunar images from the Apollo missions with respect to brightness. This difference is due to the fundamental assumptions of Mie theory, and the transfer of these equations into VR simulations. Specifically, this is due to the complexity of particles sizes and shapes (non-spherical) in the lunar exosphere. This is not accounted for in Mie calculations, for it only considers spherical particles of various sizes to exist in the atmosphere.

These exploratory mathematical manipulations led to a pivotal discovery about Mie theory and its overall accuracy in modeling light scattering on the Moon in VR. The intensity of the sun elevation the in-scattering coefficient (I'_v) for light intensity is increased to 1 and the extinction coefficients of shadows are reduced to 0. This results in the dramatic changes of light in varying degrees of sun elevation, depicted at 30, 60, and 90 degrees.

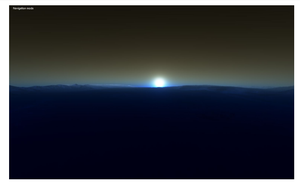


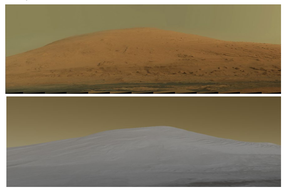
In Unity 3D, altering the denominator of Mie(G) scattering function can result in the visual representation of other cosmic phenomena, such as "virtual" black holes. This is also similar to that of a solar eclipse, or a syzygy, where the disk of the sun is fully obscured by the Moon. If you invert the phase function: *Fr*(3/4(1 +cos^2 θ)) to *Fr*(4/3(1 +cos^2 θ)), this creates a "virtual" black hole or solar eclipse.

Given that light intensity on the Moon cannot fully be rendered during the light scattering equation ambient light is increased is increased to one, so it simulates the light scattering effects of the moon while your in motion. The shader the of the extinction coefficient was raised to one to give the perception of the darkness of the space.

Rendering a Martian Atmosphere

Due to its dusty surface and reduced gravity, the atmosphere of Mars is often filled with fine dust particles, similar to those observed on the Moon. However, these particles are larger in size which is more comparable in size to the ***larger*** wavelengths of visible light, so most of the light is scattered by Mie Scattering. Visually, this results in an "inverse" scattering effect that results in a red sky and a blue sunset on the Red Planet.



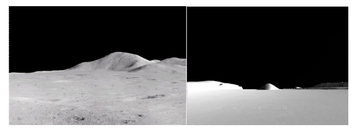


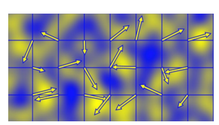
Curiosity image taken of Mars terrain (top image) and rendered simulation of Martian atmospheric scattering in VR (bottom image)

Rendering a Martian-like VE requires the adjustments of Rayleigh in-scattering coefficients to be closer to 0 and Mie coefficients closer to 1 (see figures) Collienne et al. (2013). However, what is lacking in the current images from Collienne and colleagues' (2013) current rendered Martian atmospheres is a more realistic surface texture; a similar challenge faced in the Lunar VE. Aside from the composition and toxic levels of perchlorate density in Martian soil, the dust particles on Mars create a visually similar texture to that on Earth. With this and the correct use of shaders using Perlin Noise, the next steps towards rendering realistic textures of a Martian VE is certainly feasible.

Texture, Shaders & Perlin Noise

To render an accurate representation of surface reflectance on the Moon, the visual effects of Bidirectional Reflectance Distribution Function (BDRF), also known as "non- lambertian" reflectance was rendered in Unity 3D. BDRF can cause dramatic changes in the scattering of light across the Moon's surface. The lunar regolith also lacks lambertian spectral reflectance properties causing it to appear grey in color. Theoretically, the standard Lambertian models of reflectance are different because of the lack of diffusion reflectance on the lunar surface.





Perlin Noise is an algorithm used to generate virtual dimensions and effects as seen in cinematography or VR. Gradient vectors are used to pixelate colors and add emphasis to shaders to increase the appearance of contrast. The code set for this function implements positive and negative values within tile textures to create a realistic ground plane as seen on the Apollo missions (see Figure 4). Specifically, the gradient function incorporates 12 separate images ranging from 0, -1, and 1 repeatedly to create contrast variance in the topographical gradients to mesh the tile patterns to recreate an accurate, naturalistic scene in the lunar VE. In addition, ambient light was precomputed to simulate backscattering while moving through the lunar VE. For shadows, the sun intensity was set to 0 to simulate the lack of aerial perspective on the Moon and the "blackness" of space.

$$L_o(p, \omega_o) = \int_\Omega f_r(p, \omega_i, \omega_o) L_i(p, \omega_i) \mathbf{n} \cdot \omega_i d\omega_i$$

A newer approach to simulate the veridical effects of the BDRF is Alamia's (2019) rendering equation (above). This simplified equation (as compared to Mie and Rayleigh scattering) allows the manipulation of specific points of light, while also measured at various integers. An additional concept essential to improve acuity includes the radiant flux which measures the total amount of energy emitted by a light source.

$L = \frac{d\phi^2}{dAd\omega cos}$ The radiance property (L) is defined by the combination of physical quantifiable characteristics. Phi (Φ) represents the scattering mechanism for a single ray of light.

Next, the radiant intensity represents the amount of scattered light for every solid angle. Similarly, irradiance is represented by a single point in which all the light bounces off of a certain point. $I = \frac{d\Phi}{d\omega}$

$$f_r(p, \omega_i, \omega_o) = \frac{c}{\pi}$$

Lastly, the BRDF equation measures the relationship between the incoming and outcoming light ray, and further quantifies the amount of incoming ray to final outgoing radiance. However, Alamia (2019) highlights the BDRF as: "sum of reflected light must not exceed the amount of light". The specificity of each property in the rendering equation as indicated above, illustrates the necessity of each of the light-scattering properties in relation to function. Implementation of both the rendering equation and the concept of radiant flux will help to better understand the process of applying and translating specific light scattering processes as rendered in VR.