

La Trobe University
Bundoora, Victoria, Australia
Faculty of Science, Technology & Engineering
Department of Computer Science and Computer Engineering

Sunbeams:
**An investigation into lighting and
shadows produced by the Sun and clouds**

Author:	Joseph Hura
Student No:	99527505
Course:	B. Comp. Sci. (Hons)/ B. Elec. Eng.
Submission Date:	April 2006
Supervisor:	Dr. Richard Hall

Abstract

Sunbeams are the result of the interaction between clouds, sunlight and scattering. We describe the foundations that govern the physical phenomena of sunbeams by categorising the artifacts of clouds, light and shadows. We then investigate how these artifacts have been modelled in Computer Graphics literature for physically accurate offline image synthesis. We conclude by proposing a novel method to simulate photorealistic sunbeams, by synthesising techniques and models of photorealistic offline image synthesis from Computer Graphics literature.

Acknowledgments

I would like to thank the Department of Computer Science at La Trobe University for the support they have given me in my undergraduate degree. I would also like to thank the department for providing the opportunity and resources for this Honours thesis.

I would like to thank my supervisor, Dr. Richard Hall, for his support, enthusiasm and lengthy discussions, which have guided me towards new heights.

To everyone that has helped me keep my head in the clouds for all these years, I thank-you.

Contents

Abstract	i
Acknowledgments	ii
Glossary	vii
Nomenclature	viii
1 Introduction	1
1.1 Organisation	2
2 Basic Physical Interactions Between Clouds and Sunbeams	3
2.1 Clouds	3
2.2 Interactions of Light with Clouds	5
2.3 Interactions of Shadows with Clouds	7
2.4 Summary	8
3 Physical Simulations: Cloud Modelling	10
3.1 Regular Homogeneous Techniques	10
3.2 Irregular Heterogeneous Techniques	11
3.3 Wallpaper Techniques	13
3.4 Summary	15
4 Physical Simulations: Lighting Models for Clouds	16
4.1 Irregular Techniques	16
4.2 Regular Techniques	18
4.3 Summary	20
5 Physical Simulations: Shadow Techniques	21
5.1 Radiosity	21
5.1.1 Optical Depth	22
5.1.2 Phase Function	23

5.1.3	Effects of Attenuation on Light Intensity	24
5.1.4	The Light Transport Equation	25
5.2	Summary	27
6	Conclusion	28
6.1	Research Questions	28
6.2	Implementation Plan	29
6.3	Evaluation	30
A	Perspective	31

List of Figures

1	Principal cloud forms and main sub-types	4
2	Aerosols in Clouds	4
3	Phenomena affecting light propagation	5
4	Geometry of Shadows	7
5	Optics of Sunbeams	8
6	A Voxel Grid	11
7	Deforming Metaballs	13
8	Texture Mapping	14
9	Perlin Noise	14
10	Scattering in a cloud	23
11	Qualitative Test Photograph	30
12	Perspective Traintracks	31

List of Tables

1	Hybrid Cloud Simulation Models	15
---	--	----

Glossary

2D	Two Dimensional
3D	Three Dimensional
CG	Computer Graphics
LOD	Level Of Detail
LTE	Light Transport Equation
LTM	Light Transport Method
PDE	Partial Differential Equation

Nomenclature

K_a	Absorption coefficient
K_s	Scattering coefficient
K	Extinction coefficient
ϖ_0	Single scattering albedo
$\rho(x)$	Density of a medium
τ	Optical depth of a medium
T	Transparency of a medium
α	Opacity of a medium
s	Start of a line segment
s'	End of a line segment
$\vec{\omega}$	Incident direction vector
$\vec{\omega}'$	Exiting direction vector
$\vec{x}$	Three dimensional point in space
$P(\vec{x}, \vec{\omega}, \vec{\omega}')$	Phase function
$L(\vec{x}, \vec{\omega})$	Light intensity for a given point in space

1 Introduction

Sunbeams can be defined in many ways. The Oxford Dictionary of Current English defines *sunbeams* as rays of sunlight [1]. Atmospheric science publications define sunbeams as *Crepuscular Rays* [2][3], which manifest when shadows cast by large clouds or mountains, interweaves with sunlight permeating through cloud gaps. This phenomena is commonly referred to as ‘*shafts of light*’ [4][5][6] or ‘*godrays*’ [7] in Computer Graphics (CG) literature. *For our purpose*, the definition on sunbeams is, a manifestation of shadows cast by clouds interweaved with sunlight permeating between cloud gaps.

Scattering is the result of rays of light interacting with particles in clouds and the atmosphere [8]. The heterogeneous composition of particles [9][10][11] in clouds, affects the spectral composition of light as it passes through. This process not only affects the spectral composition of light, but can also change its direction of travel. When light passes through clouds, it scatters repeatedly from particle to particle before exiting, this is termed multiple scattering.

Any graphical model of a physical phenomena must have three components [12]. The components of sunbeams are: clouds, light and shadows. Computer generated photorealistic representations of natural phenomena must take into account the physical nature of the problem [13][14][15]. An offline image synthesis approach allows for the computation of physically accurate simulations, without the simplification required for real-time approaches [16]

No specific solution for the simulation of sunbeams exists in CG literature. Ad-hoc solutions have been used to add aesthetic realism to landscapes involving clouds [4][5], or water [6][17][18]. We intend to investigate the current CG literature for methods that would be suitable for physically accurate simulations of clouds, light and shadows. We will investigate solutions for offline image synthesis of photorealistic sunbeams. Applications of our sunbeam model would be best suited to scientific or industrial visualisations, or for certain segments of the entertainment industry [19].

1.1 Organisation

In Chapter 2, we present the basic physical interactions between clouds and sunbeams. The physical properties of clouds are described, along with common types found in the environment. The physical interactions of light with clouds, taking into account scattering effects are also presented. We investigate the importance of shadows with clouds, and describe the geometry of shadows and the optics sunbeams.

In Chapter 3, we investigate physical simulations of cloud modelling in CG literature. We categorise current research into regular homogeneous techniques, irregular heterogeneous techniques and wallpaper techniques. We describe individual modelling techniques, how they have been used and how they are rendered. We discuss advantages, disadvantages and complexity issues for all techniques.

In Chapter 4, we investigate physical simulations of lighting models for clouds in CG literature. We introduce light transport methods for simulating light through clouds, by categorising current research into irregular techniques and regular techniques. We focus on high albedo and multiple scattering as our criteria for presenting heterogeneous and homogeneous approaches in both irregular and regular techniques.

In Chapter 5, we investigate physical simulations of shadow techniques in CG literature, by examining the different types of shadow simulations available in CG. We present the radiosity shadow technique and explain its application and relevance to sunbeams. Elaborating on the physical interactions of light with clouds from Chapter 2, we develop a mathematical framework for the radiative light transport equation. The framework defines cloud transparency, light scattering and the effects of attenuation on light intensity.

In Chapter 6, we summarise and present research questions from our findings. We also propose a novel approach to answering some of our research questions, and discuss our implementation evaluation plan.

2 Basic Physical Interactions Between Clouds and Sunbeams

For a physically accurate simulation of sunbeams, it is first important to understand the basic physical properties that govern this phenomenon. The interactions between the effects of light and shadows on clouds results in sunbeams. This chapter will present the basic concepts required for clouds, light and shadows in order for sunbeams to exist, and for a computer graphics simulation to be pursued.

In this chapter, clouds types and formations are discussed, along with the role aerosols play in cloud formation. The interaction between light and clouds is discussed in terms of absorption and scattering coefficients. We discuss the optical density of clouds in relation to scattering events as well as the cloud albedo. The interactions between shadows and clouds are discussed, with different shadow types and their relation to sunbeams outlined. At the end of the chapter we summarise clouds, light and shadows and tie them together to illustrate their necessity for generating sunbeams.

2.1 Clouds

Clouds form from the condensation of rising air parcels. As air parcels ascend into areas of lower pressure, they expand and cool, using their internal potential heat to supply the energy for the expansion process. Descending air parcels will encounter increasing air pressure and will experience compressional warming. Both of these processes occur at an independent, self-contained rate, termed *adiabatic*. This process leads to condensation, the formation of water droplets. The collection of water droplets combined with the ascent of air parcels leads to cloud formation.

Due to differing altitudes, clouds are categorised into three broad categories:

1. *Stratiform clouds*, which develop from warm moist air that overrides cold heavier air along a warm front.

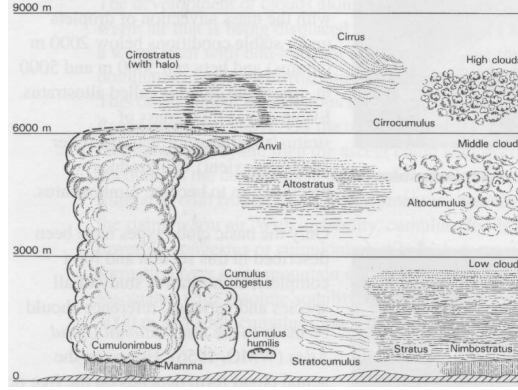


Figure 1: Principal cloud forms and main sub-types [20].

2. *Cumuliform clouds*, which develop through heat transfer from the surface and latent heat released to the air during condensation.
3. *Cirriform clouds*, which develop from the two forms of lifting stated above. They are generally found at high altitudes.

Each category contains many sub-types as shown in Figure 1, however the formation of water droplets that form clouds could not exist without atmospheric aerosols.



Figure 2: Hygroscopic aerosols act as cloud condensation nuclei which help water droplets form around them.

Atmospheric aerosols consist of many small solid and liquid particles that are either soluble in water, *hygroscopic*, *wettable* but insoluble, or water resistant, *hydrophobic* [11]. *Hygroscopic* particles best aid the condensation process by acting as cloud condensation nuclei¹. The formation of water droplets on

¹The discussion of cloud condensation nuclei is beyond the scope of this Literature Survey and further interest is referred to [11][10][21][22] for a comprehensive analysis.

aerosols is based on *Gibbs' free energy function* and *Kelvin's formula* [11]². Clouds consist of many aerosol particles and water droplets [11], with Mie Theory dictating the interaction of light with aerosol particles and cloud droplets, these are discussed in the next section.

Clouds are analogous to particle systems in CG (see Section 3). Each aerosol can be compared to a particle in particle systems, exhibiting similar characteristics of radius, density, and life-cycle.

2.2 Interactions of Light with Clouds

The path taken by a photon in the atmosphere and the interaction it makes with particles is important in the calculation of its energy. All photons that interact with a medium are either emitted, absorbed or scattered (Figure 3), whilst those that don't interact are *transmitted*[8]. Media that scatters or absorbs light is called *participating media*. In the case of photons interacting with clouds, the dominant interactions are absorption and scattering.

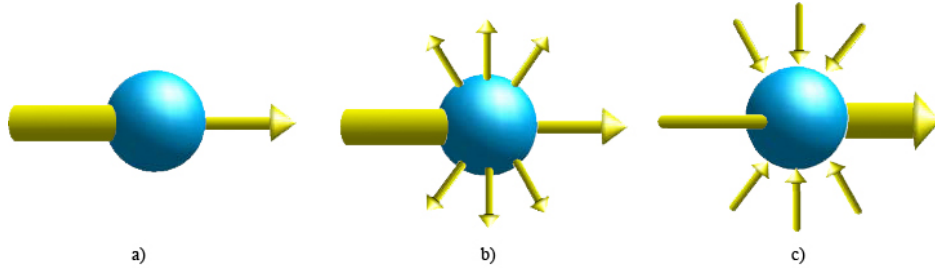


Figure 3: Phenomena affecting light propagation [23]. a) Absorption. b) Out-Scattering. c) In-Scattering.

Absorption occurs when the photon intensity is transformed into other forms of energy upon interacting with a medium, Figure 3b. In contrast, *scattering* is when a photon collides with a medium and the energy is distributed in such a way that the direction of the photon changes. Absorption and scattering attenuate the intensity of a photon as it passes through a medium. The total

²These equations are outside the scope of this Literature Survey, however, [11][10][21][22] provide a comprehensive analysis for interested readers.

attenuation of a photon is described by *extinction*. The extinction coefficient of a medium describes the amount of attenuation a photon will have when it interacts with a particle. The extinction coefficient is [24]:

$$K = K_a + K_s \quad (1)$$

where K_a and K_s are the absorption and scattering coefficients respectively. The coefficients are measured in units of inverse length, and therefore their reciprocals may be interpreted as the mean free paths for scattering and absorption. Alternatively, it may be interpreted as the probability of absorption or scattering per unit length travelled by a photon in the medium [8]. Coefficients are used to describe the optical properties of a given medium because calculating the absorption and scattering of each particle in the medium would be impractical and computationally costly. If the scattering and absorption coefficients are constant throughout the medium, we call the medium homogenous [25].

Further, $K_a = \sigma_a \rho(x)$ and $K_s = \sigma_s \rho(x)$, where $\rho(x)$ is the density of the medium, while σ_a and σ_s are the average absorption and scattering coefficients respectively. *Single scattering* is the scattering of light by a single particle in a medium, and occurs in media that are very thin or transparent. Media such as clear air are termed *optically thin*. Unlike clear air, clouds have an *optically thick* density consisting of many aerosol particles, which leads to dominant multiple scattering within the media. *Multiple scattering* is scattering of light from many particles in succession. Single scattering may be approximated using Rayleigh scattering [8], while multiple scattering may be approximated using Mie Theory [26].

Scattering may be further subdivided into two cases. The first, is light energy lost on the path of a ray in the media, which is called out-scattering (Figure 3c). The second is the light energy that has been scattered by the media in the direction of observation, this is called in-scattering (Figure 3d).

The single scattering albedo $\varpi_0 = \frac{K_s}{K_a + K_s}$, represents the fraction of light lost from scattering, while $(1 - \varpi_0)$ represents the remaining fraction which has

transformed into other forms of energy [8]. Perfect absorption occurs when $\varpi_0 = 0$, while perfect scattering is when $\varpi_0 = 1$.

The albedo of clouds is insensitive to cloud height [11] and very close to unity ($\varpi_0 \approx 1$), as absorption by water droplets is negligible in the visible spectrum [22]. Nearly all light that enters a cloud exits, but only after multiple scattering events. The dark areas in clouds are caused by the out-scattering of light rather than absorption, where the intensity of light leaving the cloud is attenuated. *Airlight* is then responsible for cloud shadows being visible in the air. Airlight is the diffuse scattered sunlight from air molecules, to differentiate it from light that is radiated from the Sun at ground level, sunlight [3].

2.3 Interactions of Shadows with Clouds

A shadow is a region of space not directly illuminated when a cloud occludes the Sun. Shadows are important as they provide useful cues for the shape and depth of objects in the natural world; hence they are inherent in the perception of realism. The *umbra* is the portion of shadow which receives no source light, whereas the *penumbra* is the portion of shadow which only partially blocks the source light. Umbrae are made visible by airlight illuminating the volume in shadow.

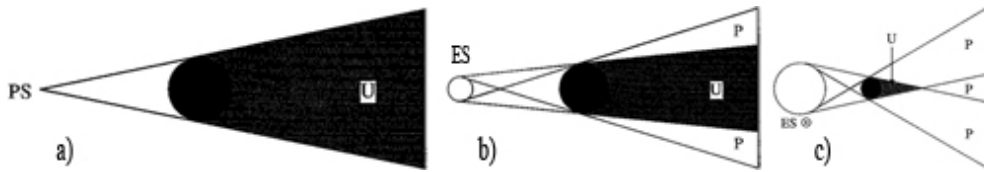


Figure 4: Geometry of Shadows [3]. a) A point source, PS , casts only an umbra U , which is totally hidden from the point source. b) An extended source, ES , like the Sun, casts both an umbra U and a penumbra P . c) When the light source like the Sun $\odot$, is physically larger than the opaque object, the umbra has a finite length. When it is smaller, the umbra extends without limit behind the opaque object as in b).

Two kinds of light sources can create shadows, *point sources* and *extended sources*. A point source is always smaller than the object it illuminates, and

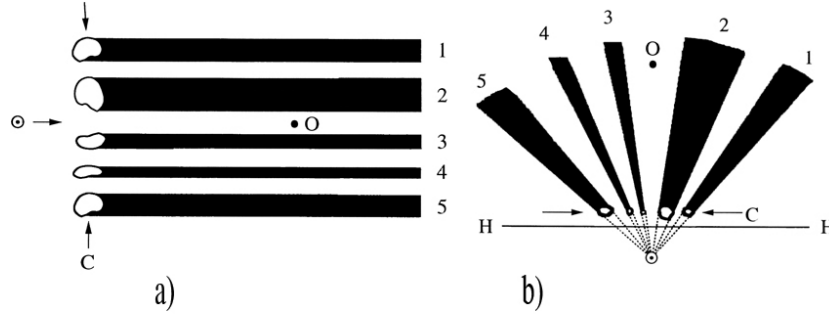


Figure 5: Optics of Sunbeams [3]. a) Looking down on the observer O from the zenith. Clouds C cast long straight parallel shadows. b) When viewed in perspective, cloud shadows seem to diverge away from the Sun ($\odot$).

the emitted rays of light diverge from a single point. Shadows cast by a point source only have an *umbra* (Figure 4a). Extended sources cast two part shadows consisting of an *umbra* and *penumbra* (Figure 4b,c). Shadows cast by an extended source, such as the Sun, emit rays of light that are near-parallel to each other; they emanate from the entire solar disc.

Cloud shadows are illuminated by airlight, while sunlight permeates between the gaps in clouds which creates a contrast between light and dark volumes in the sky, resulting in sunbeams. Sunbeams, when viewed by an observer on the ground seem to converge at the solar point, which is due to the optical process of perspective (see Appendix 5) [3].

2.4 Summary

The interweaving of shadows cast by clouds and scattered sunlight from the atmosphere form the conditions for sunbeams. A shadow is a region of space not directly illuminated when an object occludes a light source. When a shadow passes through a medium that is filled with particles able to scatter light, the shadow itself becomes visible [2]. As clouds cast shadows into the air, scattered light caused by particles in the atmosphere make it visible. Light reaching the eye due to scattering is called airlight, distinct from sunlight which is light reaching the eye directly from the Sun [27].

The combination of cloud shadows and airlight creates an effect of sunbeams converging at the solar point due to the optical process of perspective.

Creating realistic images of natural phenomena is one of the uses of Computer Graphics. In recent times, there have been major advances in photo-realistic image generation of landscapes [28][29] and natural phenomena [30][31]. Important factors when creating photo-realistic images are correct lighting and shadowing algorithms. It is of up most importance that the methods used to simulate lighting and shadows are efficient with respect to speed, they must also be aesthetically pleasing.

3 Physical Simulations: Cloud Modelling

In the previous chapter, the physical interactions between clouds and sunbeams were discussed. The physical properties of sunbeams were found to be conclusively dependant on clouds. As a result, simulating physically accurate clouds is of high importance for simulating sunbeams in Computer Graphics.

Cloud modelling in Computer Graphics (CG) can be split into three main classifications, *regular homogeneous techniques* (see Section 3.1), *irregular heterogeneous techniques* (see Section 3.2) and *wallpaper techniques* (see Section 3.3). Regular homogeneous techniques use a uniform discretisation of constant size to model clouds. Irregular heterogeneous techniques differ by specifying a volume of space with varying sized primitives. Wallpaper techniques use two dimensional (2D) images pasted onto shapes, to produce visually accurate cloud models.

3.1 Regular Homogeneous Techniques

Voxel grids are a method that is intuitive for modelling cloud densities. A voxel is the three dimensional (3D) analogy of a 2D pixel, and is a portmanteau of the words volumetric and pixel. It defines a six-sided cube in a grid subsection within a volume, and acts as a discrete sample distributed in 3D space (see Figure 6). Voxel grids are commonly used when physically-based simulations are involved [32][33][34], with volume rendering receiving much focus as a result [35][36][37]. Voxel grids have been used to store results from cloud simulation algorithms based on partial differential equations (PDEs).

Clouds were first modelled with voxel grids to store solutions from nine PDEs, which described a discretised local volume of cloud [32]. These PDEs were based on the Navier-Stokes equations for simulating fluids, and also included an equation to model the phase transition of water to vapour. A stable model of the Navier-Stokes equations [33] were used for cloud simulations with the resultant data stored on voxel grids [39]. Realistic cumulus and stratus clouds were produced with this method, however little detail about

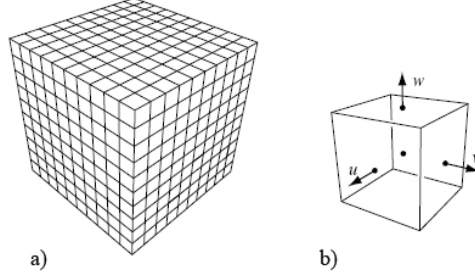


Figure 6: A Voxel Grid [38]. a) A discretised volume domain: a voxel grid. b) A volume element from the grid.

the implementation was discussed. The Navier-Stokes equations were also used to model clouds on a staggered voxel grid [40], discretised for the velocity and pressure components of the PDEs. The pressure, temperature and water content were sampled from the center of the voxels, while the velocity was defined on the faces (see Figure 6b) [38][41].

Voxel grids may be rendered using either forward or backward ray marching methods, or traditional ray tracing [32]. Advantages to using voxel grids for density distributions is that physically accurate data can be stored, producing realistic clouds when rendered. Disadvantages are in the computational cost during the rendering phase, as each cell has a complexity of $O(n^3)$; for a grid of $N_1 \times N_2 \times N_3$ the complexity is $O(n^6)$.

3.2 Irregular Heterogeneous Techniques

Particle systems intuitively model clouds, as particles best fit the model of aerosols within clouds. Particle systems model objects as clouds of primitive particles that define a volume. A particle represents a point in 3D space, with individual attributes that govern its size and colour. Particles can exist in a lifecycle, viz. particles are born and die. The model was introduced as to model fuzzy objects like clouds [42] and expanded to approximate shading, cater for self-shadowing, external shadows and external light sources [43].

Particle systems have been used to model clouds by filling space with particles of varying size and density [44]. Each particle in this method has a center, radius, density, and color. By varying the particle properties in the volume, good approximations of realistic clouds have been attained. Each particle is rendered by blending it into a frame buffer, with the transparency of a particle governing the final pixel color.

Particles have been described as the simplest surface representation in CG, enabling more primitives and complex images to be processed than polygons, in the same computation time [42]. Depending on particle properties, the cost Particles can be rendered using ray tracing methods. Advantages of producing clouds with particles is in their inherent representation of spherical volumes; less storage is required than an equivalently detailed cloud with other models. Disadvantages occur when high cloud detail is required, the number of particles increases to attain the higher detail.

Metaballs have been used to control the shape of cloud models in CG. Metaballs are also known as '*blobs*' and '*soft objects*'. Metaballs represent volumes of scalar field functions that are additive. Each metaball has a radius of influence, so when the surfaces of neighbouring metaballs intersect, the effect of each point on each surface is calculated, and the metaballs deform accordingly [45], turning into '*blobs*' (see Figure 7). Clouds can be described as big '*blobs*'. When clouds collide, they grow bigger due to coalescence of water droplets [22], intuitively, they draw similarities to metaballs. Metaball implementations for cloud models usually define each ball with a center, radius and density at the center of the ball [46][47].

A naïve implementation for clouds used metaballs to form the basic shape of a cloud [46]. Smaller metaballs were then generated recursively by using a fractal method [49] to form the fringe of the cloud. An image driven approach to model clouds from satellite images using density volumes at each ball center is another technique [47]. Wyvill's field function [50] defined the density function, and optimisations made to the radius and density of each ball, reduced error between the satellite and synthesized images.

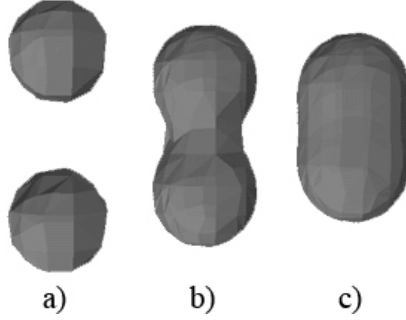


Figure 7: Deforming Metaballs [48]. a) Two metaballs. b) Deformation of points on each surface. c) The metaballs have superimposed to create one metaball.

Metaballs can be rendered a number of ways including ray-tracing, splatting [51] or using the marching cubes algorithm [52]. Advantages to using metaballs is their inherent ability to model the macrostructure of clouds, as their surfaces are smooth and malleable. Disadvantages lie in rendering, where techniques to discretise the volume are computationally high for the entire volume. Complexity for the metaball arises from the calculation of the density function when interactions occur between points on a surface, as well as the constant tracking of the field of influence.

3.3 Wallpaper Techniques

Texture mapping on the surface of ellipsoids and polygons has been used to model the visual aesthetic of clouds. Texture mapping is a technique used to paste an image onto a geometrical object like a sphere, ellipsoid or polygon (see Figure 8). Using textures reduces the computational cost needed for rendering clouds compared to methods already presented.

The visual appearance of clouds was modelled with a naïve implementation focusing on aesthetic results, using a texturing function on the surface of ellipsoids [54]. The texture function represented the spectral content of the texture pattern, similar to fractal surfaces. 2D clouds were placed as textures on a planar surface, while 3D clouds were generated with textures mapped onto the surface of ellipsoids. For added realism, the 3D ellipsoids were able to link, creating more complex cloud shapes, reminiscent of metaballs.

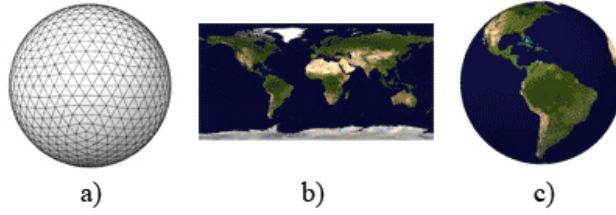


Figure 8: Texture Mapping [53]. a) Sphere with no texture. b) Texture image. c) Sphere with texture.

Texture maps are rendered using the ray tracing method. Advantages of using texture maps for cloud modelling, is the ability to economically visualise the simulation. Disadvantages lie in the inability to produce physically based cloud models. The complexity of texture maps lie in the mathematical texturing function, if it is overly complex, the advantage of using texture maps to economically visualise simulations is lost.

Procedural noise functions are able to model the wavy, soft appearance of clouds. Procedural noise techniques create realistic textures with algorithms that produce graphical representations of noise (see Figure 9) and turbulence. Noise acts as a narrow bandpass filter that has statistical invariance under rotation and translation (see Figure 9). Turbulence approximates visual appearance of turbulent flow by, approximating the magnitude of deformation from swirling around isosurfaces of the noise domain [55].

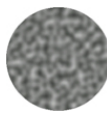


Figure 9: Perlin Noise [56].

Clouds have been successfully rendered using procedural noise techniques to fill volume densities. Turbulence was used in conjunction with a color spline to produce soft looking clouds. Clouds were modelled using turbulence to generate random, but continuous density data to fill the cloud volume [55].

Procedural noise can be rendered using standard ray tracing techniques, or by using Z-buffer or A-buffer scanline techniques [57][58]. Advantages of

a procedural noise cloud model is the ability to use nonlinear functional composition to model the stochastic aspects of clouds, as well as the flexibility of modifying the cloud appearance. Disadvantages are in the computational cost of applying noise and turbulence to large datasets, the computation for each point being $O(N^2)$.

3.4 Summary

Although regular homogeneous, irregular heterogeneous or wallpaper techniques can be used in isolation to model clouds, they have also been combined to produce hybrid models (see Table 1) for highly detailed clouds. A noticeable omission from the table, are particle systems. The omission of particle systems can be attributed to the high number of particles required to fill cloud volumes, which significantly impacts on computational cost.

Voxel Grids	Procedural Noise	Metaballs	Texture Mapping	Authors [Citation]
⊗		⊗		Dobashi [4]
⊗		⊗		Miyazaki [59]
	⊗	⊗		Ebert [60]
	⊗	⊗		Schpok [61]
		⊗	⊗	Trembilski [62]
⊗			⊗	Liao [63]

Table 1: Hybrid Cloud Simulation Models

In this chapter, cloud models were investigated in categories of regular homogeneous, irregular heterogeneous and wallpaper techniques. Wallpaper techniques do not lend themselves to physically accurate models, as light scattering due to particles can not be modelled. Irregular heterogeneous techniques best capture the physical cloud model, with particle systems representing aerosols and metaballs defining the macrostructure. Disadvantages of irregular heterogeneous or regular homogeneous techniques that involve computational cost attributed to rendering, are not of concern to our offline approach. In the next chapter, we begin our investigation of the effects of simulating light transport through clouds in CG.

4 Physical Simulations: Lighting Models for Clouds

In the previous chapter, we investigated approaches to physical cloud modelling in CG. In order to represent sunbeams we must consider methods for simulating light through participating cloud media. The light transport methods (LTMs) discussed in this chapter satisfy two requirements: a ray of light will interact with high albedo particles; the net brightness will take into account of multiple scattering effects on the ray of light [64][25]. In this chapter, we investigate two classes of techniques, *irregular* and *regular*.

Irregular LTMs have a non-uniform selection process for sampling interacting particles in the cloud, whereas regular LTMs have a uniform selection process. Further, these methods may be termed *homogeneous* or *heterogeneous* in reference to the sample space per particle. Heterogeneous methods vary the size of the sample space, while homogeneous methods keep the sample space constant.

4.1 Irregular Techniques

Irregular LTMs can be divided into two approaches, stochastic or deterministic. Stochastic approaches use a random sampling to determine solutions to the LTM. Deterministic approaches use mathematical functions that have specific solutions for defined inputs. Monte Carlo integration is an irregular homogeneous stochastic approach to solving the LTM, whereas spherical harmonics are an irregular homogeneous deterministic approach.

Monte Carlo integration is a statistical method used to solve integral equations. Monte Carlo approximations randomly sample the integral domain, alleviating the computational burden of solving the light transport equation in its entirety. Solutions can be found by intelligent sample choices, termed *importance sampling*, with the specific method depending on the problem being solved [65]. In CG, this method is known as Monte Carlo ray tracing and applying it to multiple scattering within clouds reduces the need for complex

computation. Images generated from Monte Carlo ray tracing techniques can appear noisy if not enough samples are chosen to converge on the solution.

One approach which made no restrictive assumptions about the characteristics of objects or physical phenomena in the rendering process, incorporated a unique phase function for its importance sampling [66]. The Schlick phase function was used to generate the new direction of a ray after scattering, ensuring that only the most significant scattering events determined light intensity.

Techniques similar to pure Monte Carlo ray tracing use *photon maps* [67] or *volume photon maps* [25] for calculating the LTM. These techniques provide the flexibility of pure Monte Carlo ray tracing, but are significantly more efficient, with very little noise and aliasing.

Spherical harmonics intuitively model cloud particles, as water droplets can be approximated to being spheres. Spherical harmonics $Y_l^m(\theta, \phi)$ are the angular portion of the solution to Laplace’s equation in spherical coordinates [68]. The spherical harmonics form an orthonormal basis of the functions on the unit sphere [69], meaning that a function $f(\theta, \phi)$ may be represented by an infinite series expansion:

$$f(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l P_l^m Y_l^m(\theta, \phi) \quad (2)$$

Multiple scattering in volume densities has been solved using spherical harmonics [32]. The rendering phase introduced a two pass method for calculating light scattering in volumetric media. The first pass required the scattering and absorption to be calculated along each ray and through each cloud, storing the result on voxels. The second pass traced rays from the eye and through the voxels, where the scattering of light to the eye was calculated. The two pass method has become quite common as a result of this work [66][25][70][71]. The spherical harmonic approximation yielded homogeneous matrices of PDEs which were solved by relaxation, however, only the first few spherical harmonics were used. The method was criticised by

some [72][69], as results did not accurately prove that the PDEs were solved for isotropic scattering, “since all the pictures were produced by the simpler method” [72].

Spherical harmonics have become known as the P_N -method in transport theory literature [69][73], where N is the degree of the highest harmonic in the expansion. Further research into this method observed that the general N -term expression led to a diffusion type equation, because results were truncated to the P_1 expansion [69].

The *diffusion approximation* is valid when scattering events are frequent, as in clouds [69]. If photons are travelling in random directions at any point in the medium, the multiple scattering events become apparent and the light is said to be *diffuse*. The diffusion approximation represented scattering media on a voxel grid and described two methods to solve for intensity. In the first method, the diffusion approximation was discretised to generate a system of equations which were then solved using the multi-grid method [74]. The second method was a finite element method, where a Gaussian basis function was used as a distance kernel.

4.2 Regular Techniques

Regular LTMs are generally discretised into volume elements, with each voxel representing the sample space for a light intensity at that point in space. The *discrete ordinates* method uses a collection of M discrete direction *bins*, chosen to give optimal Gaussian quadrature in the integrals over a solid angle [73]. The process however produces ray-effects, as a result of shooting energy from an element in narrow beams along the discrete directions, missing the regions between them [75]. Modifications to the equations were made [75], however, the methods share mathematical similarities to the spherical harmonics method [32]. The modified equations imply that M properly placed directions specify the directional intensity distribution to the same detail as M spherical harmonic coefficients.

For rendering of participating media with multiple anisotropic light scattering, a propagation approximation method for light scattering into M directions was proposed [72]. Two significant enhancements were presented: reduction of the ray-effect by spreading the shot radiosity into the entire direction bin; and treating multiple scattering within a single receiving element before shooting the ray again. These enhancements reduced the computational cost, but these reductions only apply to regular cubic grids and as such, the method will not work on more general finite elements.

The LTM can be solved using the *finite element method*. The method describes when an unknown function is approximated by slicing the domain of the function into a number of small elements, over which the function can be approximated using simple basis functions. The function can then be represented with a finite number of unknowns and solved numerically. *Radiosity* is a global illumination model in CG that simulates diffuse reflections among many surfaces, and often uses the finite element method to gather solutions. *Ray tracing* differs from radiosity in that it simulates light reflecting only once off each surface, but in radiosity, the surfaces of a scene represent the domain of the radiosity function.

The Zonal Method was a generalised LTM for encapsulating the radiative transfer in volumes of participating media [76]. The volume of a participating medium was divided into finite elements which were assumed to have constant radiosity. Form factors were then computed for all surface/surface, volume/volume and surface/volume pairs. Each form factor, F_{kj} , represented the ratio of the energy leaving an element, E_j , and entering an element, E_k . The algorithm calculated the factors using a depth buffer to project surfaces onto a half cube, a variation of the hemi-cube algorithm [77]. When the form factors had been found, a system of equations for the radiosities of surfaces and volumes were constructed. Complication in the method arose from surface, volume and path radiosities requiring a double integral solution. The Zonal method assumed no interference between light rays, making it limited to isotropic scattering media, thus not suitable for sunbeams.

4.3 Summary

Hybrid LTMs are also possible. One model incorporated solving integral equations, and Monte Carlo Integration [46]. The light transport between voxels was computed using the phase function as a form factor. Two simplifying observations reduced the cost of computation in this method: not all directions contributed to the illumination of a given voxel; only the first few orders of scattering contributed strongly to the illumination of a voxel.

The major issue of implementing a physically accurate multiple scattering technique for high albedo media, lies in the high computational cost attributed to solving the double integral equation for the scattering effects [40]. In CG literature, models used to represent this problem have generally been solved with simplifying assumptions that allow for certain levels of physically accurate simulations. These simplifications attempt to find a balance between a physics inspired illumination model, with high computational cost, and an aesthetically pleasing, computationally inexpensive, illumination model. Multiple scattering in CG is typically solved using *radiosity* methods originally developed in the field of thermal radiation heat transport.

In this chapter we investigated irregular and regular LTMs for high albedo participating cloud media, with focus on the multiple scattering effects that govern intensity of a light ray entering that medium. It was found that solving the light transport equation for complete multiple scattering effects is computationally costly; with a number of researchers using simplification methods to alleviate the computational burden. A majority of LTMs used a discretised volume of space, with results of light intensity stored on voxels, as a way of reducing the complexity associated with the multiple scattering problem. The computational cost of calculating the multiple scattering problem in a real-time environment would require more simplifications. As our approach is offline, we can attempt to keep simplifications to a minimum.

The next chapter will discuss shadowing simulations in CG and develop a framework for the light transport equation used in this chapter.

5 Physical Simulations: Shadow Techniques

In the previous chapter, we presented illumination models that dealt primarily with multiple scattering of participating media. Shadows are important as they provide the contrast required to distinguish sunbeams in the sky. In this chapter we will discuss how shadowing simulations in CG add inherent realism to a synthesized image.

Different types of renderable shadows exist in CG applications. Shadows may be *hard* or *soft*. Hard shadows display only the umbra (see Figure 4a) of an object. Hard shadows are determined by a binary approach, whereby a point in a scene lies in shadow of an object or not. This binary approach can be seen as either multiplying the light intensity by a 0 or 1 depending if the object lies in shadow or not. Hard shadows are generally rendered using the ray tracing method [78]. Soft shadows differ from hard shadows by including the penumbra (see Figure 4b) in image synthesis, so the binary approach of evaluation will not work. Soft shadow evaluation in its simplicity can be evaluated as the multiplication of the light intensity by a fraction in the range $[0, 1]$, where 0 indicates the umbra, 1 indicates no shadow, and all other values indicate the penumbra. The resultant shadow region has a shape that is dependant on both the occluding object and the light source [79].

To represent realistic sunbeams, a soft shadow approach must be taken. The delicate contrast between cloud shadows and sunlight permeating between cloud gaps requires this attention to detail. We will present soft shadow generation using the *radiosity* method with particular reference to the *light transport equation* (LTE).

5.1 Radiosity

Radiosity refers to the simulation of all light inter-reflections between surfaces and volumes in a scene, and is computed by solving photon interactions with all objects. Most of the methods described in Chapter 4 are forms

of the radiosity equation, where the inter-reflections between particles in a participating medium are solved using variations of the radiative LTE.

In Section 2.2 we presented the basic physical interactions between light and clouds. The interaction of light through a cloud was discussed in terms of absorption and scattering, and it was found that as light travels through a cloud, it encounters multiple scattering events due to its optical thickness. The radiative LTE for participating media describes the intensity of light distribution exiting a medium. First, a mathematical framework of the LTE must be established, by explaining its individual components in detail. The framework will elaborate on optical depth and introduce the phase function and the effects of attenuation on light intensity.

5.1.1 Optical Depth

Optical depth is a dimensionless quantity describing the *opacity* of a medium as light passes through it [45]. The total optical depth $\tau(s, s')$, of a line segment between s and s' in an inhomogeneous medium is related to the extinction coefficient K by

$$\tau(s, s') = \int_s^{s'} K(\vec{x} + t\vec{\omega}) dt \quad (3)$$

where $\vec{\omega}$ is the direction of propagating light [80][66]. An infinite optical depth for a medium means it is opaque.

The *transparency* of a medium is derived from optical depth, and is the percentage of light that leaves a point at s and reaches a point on s' along the line segment.

$$T(s, s') = e^{-\tau(s, s')} \quad (4)$$

Transparency is a more useful concept for clouds, therefore the optical depth of a segment can be described as the inverse of transparency, $\alpha(s, s') = 1 - T(s, s')$ [73].

5.1.2 Phase Function

Upon entering the cloud, incoming light undergoes a series of scattering and absorption events that modify both the directional structure of the incoming light field and its intensity. As a result of multiple scattering events, the original intensity distribution undergoes angular, spatial and temporal spreading which results in a different intensity distribution (see Figure 10).

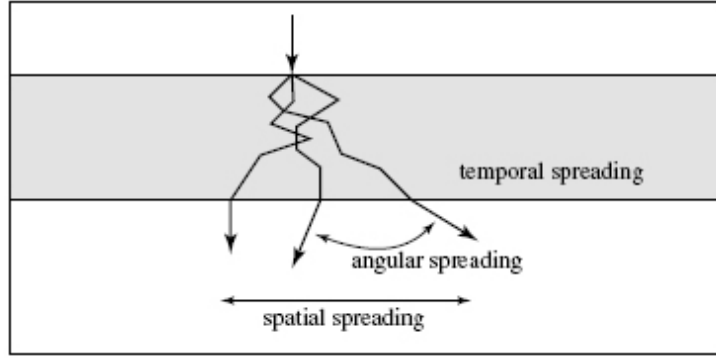


Figure 10: Scattering in a cloud [24]. The original intensity undergoes a series of scattering events that result in angular, spatial and temporal spreading of the original intensity distribution.

The phase function describes the probability density of light coming from direction $\vec{\omega}$ and scattering into direction $\vec{\omega}'$. The phase function is normalised so that $\int_{4\pi} P(\vec{\omega}, \vec{\omega}') d\omega' = 1$, and depends only on the phase angle, $\cos(\theta) = \vec{\omega} \cdot \vec{\omega}'$ [81].

The following phase functions are representative of spherical particles, which suit their application to clouds, as water droplets are spherical.

Isotropic Phase Function

An isotropic phase function scatters light equally in all directions [66].

$$P_I(\phi) = 1 \tag{5}$$

Rayleigh-Gans Phase Function

When particles are small compared to the wavelength of incident light, the phase function is given by [66]

$$P_{RG}(\phi) = \frac{3}{4} \frac{(1 + \cos^2 \phi)}{\lambda^4} \quad (6)$$

Mie Phase Function

When particles are large compared to the wavelength of incident light, the phase function is defined by Mie theory. A common formula for approximating the Mie scattering for spherical particles is the Henyey-Greenstein phase function [66][82][83][84].

$$P_{HG}(\phi) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g\cos\phi)^{3/2}} \quad (7)$$

Although these phase functions can be used for clouds, not all can be applied for a physically accurate model. The anisotropic nature of clouds requires either the Rayleigh-Gans or the Henyey-Greenstein phase functions. Taking into account that the nature of particles inside clouds are aerosols, where particles are larger than the wavelength of incident light, the scattering anisotropy will be modelled using the Henyey-Greenstein phase function [83].

5.1.3 Effects of Attenuation on Light Intensity

Remember that the extinction coefficient is the sum of all absorption and scattering, $K = K_a + K_s$, it describes the total amount of attenuation to a photon's intensity per unit length travelled through a medium. We are then able to calculate the effects of the attenuation of light intensity in a cloud.

Absorption

The absorption coefficient, K_a is the probability of absorption per unit length, therefore the change in intensity dL due to absorption over a distance ds in direction $\vec{\omega}$ is [25]

$$\frac{dL(\vec{x}, \vec{\omega})}{ds} = -K_a(\vec{x})L(\vec{x}, \vec{\omega}) \quad (8)$$

Out-Scattering

The scattering coefficient, K_s is used to calculate the out-scattering, and is similar to absorption [25]. The change in intensity dL due to scattering K_s ,

over a distance ds in direction $\vec{\omega}$ is

$$\frac{dL(\vec{x}, \vec{\omega})}{ds} = -K_s(\vec{x})L(\vec{x}, \vec{\omega}) \quad (9)$$

Extinction

Because $K = K_a + K_s$, equations 8 and 9 are effectively combined into a single equation [25].

$$\frac{dL(\vec{x}, \vec{\omega})}{ds} = -K(\vec{x})L(\vec{x}, \vec{\omega}) \quad (10)$$

In-Scattering

The in-scattered intensity of a particle depends on the amount of scattering and the direction of scattering. The phase function is used to determine how much of the scattered light at $\vec{x}$ is scattered in the direction $\vec{\omega}$. As light from any direction may be scattered into direction $\vec{\omega}$, it is important to calculate the intensity of incoming directions over the entire particle. Therefore, the change in intensity dL due to in-scattering over a distance ds in direction $\vec{\omega}$ is

$$\frac{dL(\vec{x}, \vec{\omega})}{ds} = K_s(\vec{x}) \int_{4\pi} P(x, \vec{\omega}, \vec{\omega}') L(\vec{x}, \vec{\omega}') d\vec{\omega}' \quad (11)$$

where $\vec{\omega}'$ is the incoming direction of in-scattered light [25].

5.1.4 The Light Transport Equation

The direction $\vec{\omega}$, and intensity of light $L(\vec{x}, \vec{\omega})$, change as a result of a ray of light passing through a cloud. Intensity of light is attenuated due to absorption and out-scattering (Figure 3a,b), while in-scattering intensifies light (Figure 3c) [8][85].

$$\frac{dL(\vec{x}, \vec{\omega})}{ds} = -K(\vec{x})L(\vec{x}, \vec{\omega}) + K_s(\vec{x}) \int_{4\pi} P(\vec{x}, \vec{\omega}, \vec{\omega}') L(\vec{x}, \vec{\omega}') d\vec{\omega}' \quad (12)$$

This equation can be solved by bringing the extinction term to the left hand side and multiplying by the integrating factor [73]

$$\exp\left(\int_0^s K(t)dt\right)$$

If a ray is parameterised in terms of $t \in [0, D]$, where $t = 0$ is the point on the medium where the light is incident, and $t = D$ where the light exits the medium, equation 12 can be integrated to find the exitant intensity at $t = D$ [73].

$$L(D, \vec{\omega}) = T(0, D)L(0, \vec{\omega}) + \int_0^D T(s, D)g(s)ds \quad (13)$$

where $L(0, \vec{\omega})$ is the incident light intensity, $T(s, D)$ is the transparency defined in equation 4 and $g(s)$ is

$$g(s) = K_s(\vec{x}(s)) \int_{4\pi} P(\vec{x}, \vec{\omega}, \vec{\omega}') L(\vec{x}(s), \vec{\omega}') d\omega \quad (14)$$

Equation 13 is similar to the classic volume rendering model [86], where the $g(s)$ term is

$$g(s) = R(x(s))f_s(x(s))L_d \quad (15)$$

where $R(x(s))$ is the surface reflectivity color, $f_s(x(s))$ is the Blinn-Phong surface shading model, and L_d is the intensity of a point light source.

Equation 13 is important for cloud illumination, as the first term represents the *extinction term* and, the second term represents the *in-scattering term*. The extinction term is the light intensity coming from the background and reaching the viewer. The in-scattering term represents light scattered into the view direction from all particles within the cloud.

Using this model, an accurate multiple scattering solution is computationally expensive. To calculate the intensity of light scattered to a point $\vec{p}_i$ in the cloud from another point $\vec{p}_k$, the intensity of light at $\vec{p}_k$ must first be determined. The calculation is itself recursive, as the intensity also depends on

absorption and scattering. The techniques described in Chapter 4 propose approximate solutions to this problem.

Shadows are inherently generated using the LTE as a radiosity approach. The light intensity is attenuated as it interacts with particles in the clouds and air, with each interaction having an effect on the final light intensity reaching the viewer.

5.2 Summary

In this chapter, hard and soft shadows were evaluated for their use with sunbeams. Soft shadows were deemed more appropriate to simulate the delicate contrast between cloud shadows and sunlight permeating between cloud gaps. The radiosity technique used to apply soft shadows to the current problem was discussed in terms of the radiative LTE for participating media. The mathematical framework of light interaction with clouds discussed in Section 2.2 was extended, to include optical depth, the phase function, and the effect of attenuation on light intensity, in order to better understand the LTE. The LTE is useful as a soft shadow approach as light intensity reaching the viewer, and shadow computation, is inherent in its derivation. The multiple scattering problem was found to be recursive with a high computational cost for higher order multiple scattering events. The impact in a real-time application would be significant, but in our offline image synthesis approach, the impact is negligible. In the next chapter, we draw our conclusions on the research conducted by formulating possible research questions and a proposal for an implementation plan of sunbeams.

6 Conclusion

In the previous chapter, we presented a radiosity approach to create delicate contrasts in an offline image synthesis of sunbeams. In this chapter, we draw our conclusions on the research conducted into the fields of cloud modelling, light transport methods and shadow techniques in CG, by *proposing research questions* that we feel were not addressed by CG literature. We also present a novel method for answering some of the questions we have raised. We conclude this chapter by discussing evaluation techniques that will be used to test our novel approach.

6.1 Research Questions

Our conclusions are summarised into a series of research questions that demonstrate our findings of the current gaps in CG research. The research questions also address the current problem of generating a photorealistic image of sunbeams.

- *Can a physically accurate hybrid cloud model utilising Particle Systems be created?*
- *Can a physically accurate cloud model be formed from Irregular Heterogeneous techniques?*
- *Can the Schlick phase function be incorporated into other light transport models?*
- *Can multiple scattering events in clouds be minimised, without losing the visual qualities of a photorealistic image synthesis of sunbeams?*
- *Can the LTE equation be modified to suit a direct application to sunbeams?*

6.2 Implementation Plan

Our novel approach will implement an offline, physically accurate, image synthesis of sunbeams. We will need to generate a scene matching our reference photo (see Figure 11), with a scene descriptor assisting the placement of objects. To generate the convergent effects of sunbeams, we will use a perspective camera projection onto the image plane.

Our scene inputs will be a light source and clouds. We will use a directional light source, as the Sun's rays emit near parallel rays, with the position and intensity its only attributes. The cloud positions will also be expressed in the scene descriptor.

Our *novel implementation* will attempt to answer the following questions:

- *Can a physically accurate cloud model be formed from Irregular Heterogeneous techniques?*
- *Can the Schlick phase function be incorporated into other light transport models?*
- *Can we generate photorealistic sunbeams?*

Irregular heterogeneous techniques will be used to model clouds, utilising particle systems as they closely resemble aerosols. We will investigate if particle systems are the best suited cloud modelling representation for this application. To investigate *light transport methods*, we will use a Monte Carlo ray tracing technique incorporating the Schlick phase function [66]. As a secondary exercise, will attempt to incorporate the Schlick phase function with other light transport methods, such as discrete ordinates or the finite element method.

We will render our images using *radiosity* methods and equations 13 and 14, as the basis for our lighting and shadowing algorithm. The output should match our reference photo.

6.3 Evaluation

Evaluation of our novel implementation will use a mixture of *qualitative* and *quantitative* approaches. Qualitative approaches will be comparisons of our synthesised image to a real photograph of sunbeams (see Figure 11). Qualitatively, the contrast between cloud shadows and the sunlight permeating between the gaps in clouds must be the same on visual inspection. Image processing tools will allow us to quantitatively analyse the synthesised image to the real photograph. As our implementation does not focus on rendering time, our approach is to get the best match for each different technique used.



Figure 11: Qualitative Test Photograph. The synthesised image produced by our approach will be compared to this photograph.

A Perspective

From Section 2.3, emitted rays of light from the Sun are near parallel. With visual perspective, the eyes refocus near-parallel rays arriving from distant objects toward a focal point within the eye, the cornea. Our brain can only process the angular size of an object by analysing its angular measurement within our field of view [87].

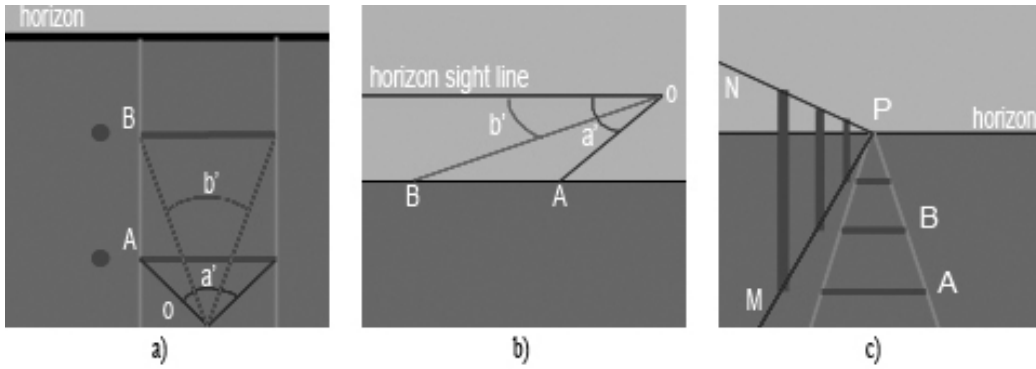


Figure 12: Perspective Traintracks. a) Top View: The angle created by b' is smaller than that of a' , so B appears smaller than A . b) Side View. The angle created by b' is smaller than that of a' to the horizon sight line. c) Observer's View: All parallel lines extending to the horizon converge to P , the vanishing point.

With reference to Figure 12, a classic train track scenario is illustrated to explain perspective [87]. The scenario consists of straight railroad tracks stretching to the horizon, as well as telephone poles on one side of the tracks. The train tracks, tie points and telephone poles are equidistant and the telephone poles are all of the same height. As these objects approach the horizon, the tracks converge to a single point, while the distance between tie points decreases and the telephone poles diminish in size.

Figure 12a shows the observer's position O , in the middle of a train track, with tie points A and B being equidistant. From the observer's position, the angle created by b' is smaller than that of a' therefore B will appear smaller than A .

Figure 12b helps determine the relative placement of objects within an observer's field of view. The angle of declination of point B is smaller than

that of A with respect to the angle of declination to the horizon (0°). Point A will then appear below point B , with both being below the horizon line.

Figure 12c is the observer's view of the railway tracks as a combination of the previous two images. The tops and bottoms of the perpendicular telephone poles are joined by invisible orthogonal lines, M and N , that extend to point P . All parallel lines within the scenario extend to the horizon, converging at point P , the vanishing point. This is called perspective.

References

- [1] C. Soanes and S. Hawker, eds., *Compact Oxford English Dictionary of Current English*. Oxford University Press, third ed., June 2005.
- [2] J. Naylor, *Out of the Blue: A 24-hour Skywatcher's Guide*. Cambridge University Press, 2002.
- [3] D. K. Lynch and W. Livingston, *Color and Night in Nature*. Cambridge University Press, second ed., 2001.
- [4] Y. Dobashi, K. Kaneda, H. Yamashita, T. Okita, and T. Nishita, “A simple, efficient method for realistic animation of clouds,” in *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 19–28, ACM Press/Addison-Wesley Publishing Co., 2000.
- [5] Y. Dobashi, T. Yamamoto, and T. Nishita, “Interactive rendering of atmospheric scattering effects using graphics hardware,” in *HWWS '02: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, (Aire-la-Ville, Switzerland), pp. 99–107, Eurographics Association, 2002.
- [6] K. Iwasaki, T. Nishita, and Y. Dobashi, “Efficient rendering of optical effects within water using graphics hardware,” in *PG '01: Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*, (Washington DC, USA), p. 374, IEEE Computer Society, 2001.
- [7] K. Hegeman, M. Ashikhmin, and S. Premože, “A lighting model for general participating media,” in *SI3D '05: Proceedings of the 2005 symposium on Interactive 3D graphics and games*, (New York, USA), pp. 117–124, ACM Press, 2005.
- [8] S. Chandrasekhar, *Radiative Transfer*. New York, USA: Dover Publications, Inc., 1960.
- [9] J. Sloup, “A survey of the modelling and rendering of the earth's atmosphere,” in *SCCG '02: Proceedings of the 18th spring conference on Computer graphics*, (New York, USA), pp. 141–150, ACM Press, 2002.
- [10] J. V. Iribarne and H.-R. Cho, *Atmospheric Physics*. D. Reidel Publishing Company, 1980.

- [11] M. L. Salby, *Fundamentals of Atmospheric Physics*. Academic Press, 1996.
- [12] J. Stam and E. Fiume, “Depicting fire and other gaseous phenomena using diffusion processes,” in *SIGGRAPH ’95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 129–136, ACM Press, 1995.
- [13] X. D. He, K. E. Torrance, F. X. Sillion, and D. P. Greenberg, “A comprehensive physical model for light reflection,” *SIGGRAPH Comput. Graph.*, vol. 25, no. 4, pp. 175–186, 1991.
- [14] J. T. Kajiya, “Anisotropic reflection models,” in *SIGGRAPH ’85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 15–21, ACM Press, 1985.
- [15] B. Sosorbaram, T. Fujimoto, K. Muraoka, and N. Chiba, “Visual simulation of lightning taking into account cloud growth,” in *CGI ’01: Proceedings of the International Conference on Computer Graphics*, (Washington DC, USA), p. 89, IEEE Computer Society, 2001.
- [16] J. T. Kajiya, “The rendering equation,” in *SIGGRAPH ’86: Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 143–150, ACM Press, 1986.
- [17] T. Nishita and E. Nakamae, “Method of displaying optical effects within water using accumulation buffer,” in *SIGGRAPH ’94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 373–379, ACM Press, 1994.
- [18] H. W. Jensen, “Underwater sunbeams,” in *SIGGRAPH ’98: ACM SIGGRAPH 98 Electronic art and animation catalog*, (New York, USA), p. 180, ACM Press, 1998.
- [19] J. Hura and R. Hall, “Design of a simulation of atmospheric sunbeams,” *WSEAS Transactions on Computers*, vol. 5, no. 10, pp. 2466–2471, 2006.
- [20] R. D. Thompson, *Atmospheric Processes and Systems*. Routledge, 1998.
- [21] R. G. Fleagle and J. A. Businger, *An Introduction to Atmospheric Physics*. Academic Press, second ed., 1980.
- [22] R. Goody, *Principles of Atmospheric Physics and Chemistry*. Oxford University Press, 1995.

- [23] A. Boudet, P. Pitot, D. Pratumarty, and M. Paulin, “Photon splatting for participating media,” in *GRAPHITE '05: Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, (New York, USA), pp. 197–204, ACM Press, 2005.
- [24] S. Premože, M. Ashikhmin, and P. Shirley, “Path integration for light transport in volumes,” in *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, (Aire-la-Ville, Switzerland), pp. 52–63, Eurographics Association, 2003.
- [25] H. W. Jensen and P. H. Christensen, “Efficient simulation of light transport in scenes with participating media using photon maps,” in *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 311–320, ACM Press, 1998.
- [26] H. van de Hulst, *Light Scattering by Small Particles*. New York, USA: Dover Publications, Inc., 1981.
- [27] B. Sun, R. Ramamoorthi, S. G. Narasimhan, and S. K. Nayar, “A practical analytic single scattering model for real time rendering,” *ACM Trans. Graph.*, vol. 24, no. 3, pp. 1040–1049, 2005.
- [28] H. W. Jensen, F. Durand, J. Dorsey, M. M. Stark, P. Shirley, and S. Premože, “A physically-based night sky model,” in *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 399–408, ACM Press, 2001.
- [29] A. J. Preetham, P. Shirley, and B. Smits, “A practical analytic model for daylight,” in *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 91–100, ACM Press/Addison-Wesley Publishing Co., 1999.
- [30] R. V. Klassen, “Modeling the effect of the atmosphere on light,” *ACM Trans. Graph.*, vol. 6, no. 3, pp. 215–237, 1987.
- [31] G. V. G. Baranoski, J. Wan, J. G. Rokne, and I. Bell, “Simulating the dynamics of auroral phenomena,” *ACM Trans. Graph.*, vol. 24, no. 1, pp. 37–59, 2005.
- [32] J. T. Kajiya and B. P. V. Herzen, “Ray tracing volume densities,” in *SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 165–174, ACM Press, 1984.

- [33] J. Stam, “Stable fluids,” in *SIGGRAPH ’99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 121–128, ACM Press/Addison-Wesley Publishing Co., 1999.
- [34] F. Losasso, F. Gibou, and R. Fedkiw, “Simulating water and smoke with an octree data structure,” *ACM Trans. Graph.*, vol. 23, no. 3, pp. 457–462, 2004.
- [35] R. A. Drebin, L. Carpenter, and P. Hanrahan, “Volume rendering,” in *SIGGRAPH ’88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 65–74, ACM Press, 1988.
- [36] L. M. Sobierajski and A. E. Kaufman, “Volumetric ray tracing,” in *VVS ’94: Proceedings of the 1994 symposium on Volume visualization*, (New York, USA), pp. 11–18, ACM Press, 1994.
- [37] S. Roettger, S. Guthe, D. Weiskopf, T. Ertl, and W. Strasser, “Smart hardware-accelerated volume rendering,” in *VISSYM ’03: Proceedings of the symposium on Data visualisation 2003*, (Aire-la-Ville, Switzerland), pp. 231–238, Eurographics Association, 2003.
- [38] R. Fedkiw, J. Stam, and H. W. Jensen, “Visual simulation of smoke,” in *SIGGRAPH ’01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 15–22, ACM Press, 2001.
- [39] D. Overby, Z. Melek, and J. Keyser, “Interactive physically-based cloud simulation,” in *PG ’02: Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, (Washington DC, USA), p. 469, IEEE Computer Society, 2002.
- [40] M. J. Harris, W. V. Baxter, T. Scheuermann, and A. Lastra, “Simulation of cloud dynamics on graphics hardware,” in *HWWS ’03: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, (Aire-la-Ville, Switzerland), pp. 92–101, Eurographics Association, 2003.
- [41] N. Foster and D. Metaxas, “Modeling the motion of a hot, turbulent gas,” in *SIGGRAPH ’97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 181–188, ACM Press/Addison-Wesley Publishing Co., 1997.

- [42] W. T. Reeves, "Particle systems a technique for modeling a class of fuzzy objects," *ACM Trans. Graph.*, vol. 2, no. 2, pp. 91–108, 1983.
- [43] W. T. Reeves and R. Blau, "Approximate and probabilistic algorithms for shading and rendering structured particle systems," in *SIGGRAPH '85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 313–322, ACM Press, 1985.
- [44] M. J. Harris and A. Lastra, "Real-time cloud rendering," in *EG 2001 Proceedings* (A. Chalmers and T.-M. Rhyne, eds.), vol. 20 of 3, pp. 76–84, Blackwell Publishing, 2001.
- [45] J. F. Blinn, "A generalization of algebraic surface drawing," *ACM Trans. Graph.*, vol. 1, no. 3, pp. 235–256, 1982.
- [46] T. Nishita, Y. Dobashi, and E. Nakamae, "Display of clouds taking into account multiple anisotropic scattering and sky light," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 379–386, ACM Press, 1996.
- [47] Y. Dobashi, T. Nishita, H. Yamashita, and T. Okita, "Modeling of clouds from satellite images using metaballs," in *PG '98: Proceedings of the 6th Pacific Conference on Computer Graphics and Applications*, (Washington DC, USA), pp. 53–60, IEEE Computer Society, 1998.
- [48] W. T. F. Encyclopedia, "Image:metaballs.gif." <http://en.wikipedia.org/wiki/Image:Metaballs.gif>.
- [49] A. Fournier, D. Fussell, and L. Carpenter, "Computer rendering of stochastic models," *Commun. ACM*, vol. 25, no. 6, pp. 371–384, 1982.
- [50] G. Wyvill and A. Trotman, "Ray-tracing soft objects," in *CG International '90: Proceedings of the eighth international conference of the Computer Graphics Society on CG International '90: computer graphics around the world*, (New York, USA), pp. 469–476, Springer-Verlag New York, Inc., 1990.
- [51] L. Westover, "Footprint evaluation for volume rendering," in *SIGGRAPH '90: Proceedings of the 17th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 367–376, ACM Press, 1990.

- [52] W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3d surface construction algorithm,” in *SIGGRAPH ’87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 163–169, ACM Press, 1987.
- [53] W. T. F. Encyclopedia, “Image:texturemapping.png.” <http://en.wikipedia.org/wiki/Image:TextureMapping.png>.
- [54] G. Y. Gardner, “Visual simulation of clouds,” in *SIGGRAPH ’85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 297–304, ACM Press, 1985.
- [55] K. Perlin, “An image synthesizer,” in *SIGGRAPH ’85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 287–296, ACM Press, 1985.
- [56] K. Perlin, “Making noise. based on a talk presented at gdchardcore (dec 9, 1999).” <http://www.noisemachine.com/talk1/6.html>.
- [57] L. Carpenter, “The a -buffer, an antialiased hidden surface method,” in *SIGGRAPH ’84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 103–108, ACM Press, 1984.
- [58] D. S. Ebert and R. E. Parent, “Rendering and animation of gaseous phenomena by combining fast volume and scanline a-buffer techniques,” in *SIGGRAPH ’90: Proceedings of the 17th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 357–366, ACM Press, 1990.
- [59] R. Miyazaki, S. Yoshida, T. Nishita, and Y. Dobashi, “A method for modeling clouds based on atmospheric fluid dynamics,” in *PG ’01: Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*, (Washington DC, USA), p. 363, IEEE Computer Society, 2001.
- [60] O. Deussen, D. S. Ebert, R. Fedkiw, F. K. Musgrave, P. Prusinkiewicz, D. Roble, J. Stam, and J. Tessendorf, “The elements of nature: interactive and realistic techniques,” in *GRAPH ’04: Proceedings of the conference on SIGGRAPH 2004 course notes*, (New York, USA), p. 32, ACM Press, 2004.
- [61] J. Schpok, J. Simons, D. S. Ebert, and C. Hansen, “A real-time cloud modeling, rendering, and animation system,” in *SCA ’03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer*

- animation*, (Aire-la-Ville, Switzerland), pp. 160–166, Eurographics Association, 2003.
- [62] A. Trembilski and A. Broßler, “Transparency for polygon based cloud rendering,” in *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*, (New York, USA), pp. 785–790, ACM Press, 2002.
 - [63] H.-S. Liao, J.-H. Chuang, and C.-C. Lin, “Efficient rendering of dynamic clouds,” in *VRCAI '04: Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, (New York, USA), pp. 19–25, ACM Press, 2004.
 - [64] J. F. Blinn, “Light reflection functions for simulation of clouds and dusty surfaces,” in *SIGGRAPH '82: Proceedings of the 9th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 21–29, ACM Press, 1982.
 - [65] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. New York, USA: Springer-Verlag, 1999.
 - [66] P. Blasi, B. L. Saëc, and C. Schlick, “A rendering algorithm for discrete volume density objects,” *Computer Graphics Forum (Eurographics '93)*, vol. 12, no. 3, pp. 201–210, 1993.
 - [67] H. W. Jensen, “Global illumination using photon maps,” in *Proceedings of the eurographics workshop on Rendering techniques '96*, (London, UK), pp. 21–30, Springer-Verlag, 1996.
 - [68] G. B. Arfken, *Mathematical Methods for Physicists*. Academic Press, third ed., 1985.
 - [69] J. Stam, “Multiple scattering as a diffusion process,” in *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)* (P. M. Hanrahan and W. Purgathofer, eds.), (New York, NY), pp. 41–50, Springer-Verlag, 1995.
 - [70] J. Kniss, S. Premože, C. Hansen, and D. Ebert, “Interactive translucent volume rendering and procedural modeling,” in *VIS '02: Proceedings of the conference on Visualization '02*, (Washington DC, USA), pp. 109–116, IEEE Computer Society, 2002.
 - [71] G. Schussman and K.-L. Ma, “Anisotropic volume rendering for extremely dense, thin line data,” in *VIS '04: Proceedings of the conference on Visualization '04*, (Washington, DC, USA), pp. 107–114, IEEE Computer Society, 2004.

- [72] N. L. Max, “Efficient light propagation for multiple anisotropic volume scattering,” in *Fifth Eurographics Workshop on Rendering*, (Darmstadt, Germany), pp. 87–104, 1994.
- [73] N. L. Max, “Optical models for direct volume rendering,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 1, no. 2, pp. 99–108, 1982.
- [74] W. Hackbush, *Multi-grid Methods and Applications*. Berlin, Germany: Springer-Verlag, second ed., 2003.
- [75] K. D. Lathrop, “Ray effects in discrete ordinates equations,” in *Nuclear Science and Engineering*, vol. 32, pp. 357–369, 1968.
- [76] H. E. Rushmeier and K. E. Torrance, “The zonal method for calculating light intensities in the presence of a participating medium,” in *SIGGRAPH ’87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 293–302, ACM Press, 1987.
- [77] M. F. Cohen and D. P. Greenberg, “The hemi-cube: a radiosity solution for complex environments,” in *SIGGRAPH ’85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 31–40, ACM Press, 1985.
- [78] T. Whitted, “An improved illumination model for shaded display,” *Commun. ACM*, vol. 23, no. 6, pp. 343–349, 1980.
- [79] A. Woo, P. Poulin, and A. Fournier, “A survey of shadow algorithms,” *IEEE Comput. Graph. Appl.*, vol. 10, no. 6, pp. 13–32, 1990.
- [80] K. Riley, D. Ebert, C. Hansen, and J. Levit, “Visually accurate multi-field weather visualization,” in *VIS ’03: Proceedings of the 14th IEEE Visualization 2003 (VIS’03)*, (Washington DC, USA), p. 37, IEEE Computer Society, 2003.
- [81] S. Premože, “Analytic light transport approximations for volumetric materials,” in *PG ’02: Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, (Washington DC, USA), p. 48, IEEE Computer Society, 2002.
- [82] N. L. Max, G. Schussman, R. Miyazaki, K. Iwasaki, and T. Nishita, “Diffusion and multiple anisotropic scattering for global illumination in clouds,” in *Journal of WSCG*, vol. 12 of 3, pp. 277–284, 2004.

- [83] J. Haber, M. Magnor, and H.-P. Seidel, “Physically-based simulation of twilight phenomena,” *ACM Trans. Graph.*, vol. 24, no. 4, pp. 1353–1373, 2005.
- [84] D. Q. Nguyen, R. Fedkiw, and H. W. Jensen, “Physically based modeling and animation of fire,” in *SIGGRAPH ’02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, (New York, USA), pp. 721–728, ACM Press, 2002.
- [85] J. Arvo, “Transfer functions in global illumination,” in *ACM SIGGRAPH ’93 Course Notes - Global Illumination*, pp. 1–28, ACM Press, 1993.
- [86] K. Engel, M. Hadwiger, J. M. Kniss, A. E. Lefohn, C. R. Salama, and D. Weiskopf, “Real-time volume graphics,” in *GRAPH ’04: Proceedings of the conference on SIGGRAPH 2004 course notes*, (New York, USA), p. 266, ACM Press, 2004.
- [87] M. A. Penna and R. R. Patterson, *Projective Geometry And Its Applications To Computer Graphics*. Prentice-Hall, 1986.