

UTRECHT UNIVERSITY

MASTER THESIS

GRADUATE SCHOOL OF NATURAL SCIENCES

**Efficient Lightcuts for Virtual Ray Lights through
Non-linear Media**

Author

Vincent J.S. VERSNEL

Supervisor

Dr. Jacco BIKKER

Second Supervisor

Dr. Amir VAXMAN

July 12, 2022

1 Acknowledgements

First of all, I would like to express my sincere gratitude towards my supervisor Jacco Bikker for enabling me to engage with computer graphics research and for guiding me along the way. I am also grateful to Wojciech Jarosz for sharing their implementation which significantly aided in the implementation of the experiments for this master thesis.

2 Abstract

Non-linear media causes many interesting physically based effects in nature by bending the trajectory of photons. Furthermore, participating media causes light to scatter many times resulting in a spreading of the light. Rendering multiple scattering in a non-linear environment has been solved with the use of photon mapping [9] by using it to compute both direct and indirect illumination. However, photon mapping requires millions to billions of photons to be traced during the preprocessing stage. Throughout recent years, multiple scattering has been solved more efficiently using higher dimensional light samples. Unfortunately, recent work in non-linear media limits itself to single scattering. The aim of this research is to combine recent advancements in efficiently rendering multiple scattering within a non-linear environment. We show that with slight modifications we can use virtual ray lights with lightcuts in a non-linear environment which significantly reduces the total computation time and memory footprint. Furthermore, we show that with the application of stochastic lightcuts the method has potential for interactive applications.

Contents

1	Acknowledgements	1
2	Abstract	2
3	Introduction	5
3.1	Contribution	5
4	Motivation	6
5	Preliminaries	7
5.1	Participating Media	7
5.1.1	Radiative Transfer Equation	7
5.1.2	Volume Rendering Equation	8
5.1.3	Phase Function	8
5.1.4	Transmittance	8
5.2	Refraction	9
5.2.1	Law of refraction	9
5.2.2	Refractive Radiative Transfer Equation	10
6	Literature Study	12
6.1	Multiple scattering	12
6.1.1	Virtual Point Lights	12
6.1.2	Virtual Ray Lights	12
6.1.3	Virtual Beam Lights	14
6.1.4	Many-light Optimizations	14
6.1.5	Volumetric Photon Mapping	16
6.1.6	Photon Beams	16
6.1.7	ReSTIR	16
6.2	Non-linear Ray Tracing	17
6.2.1	General Methods	17
6.2.2	Atmospheric Phenomena	18
6.2.3	Gravitational lensing	18
7	Research Methodology	19
7.1	Research Aims	19
7.2	Method	19
8	Implementation	20
8.1	Integrators	20
8.1.1	Volumetric Photon Mapping	20
8.1.2	VRL	20
8.1.3	VRL Lightcuts	20
8.1.4	Stochastic Lightcuts	21
8.1.5	Non-linear Media	21

9 Results	23
9.1 Cornell Box Scene	25
9.2 Laser Scene	26
9.3 Mirage Scene	29
10 Discussion and Limitations	31
11 Conclusion	32
12 Future Work	32
13 Appendix	36

3 Introduction

State-of-the-art ray tracers often assume a linear environment where photons travel in straight lines and only change direction at surface boundaries of objects or media. However, in the real world there are many participating media that are heterogeneous where light paths are bent continuously throughout the media. Two primary examples on earth are the atmosphere and the ocean. Spatial variations in temperature, pressure, plankton, humidity, and much more affect the refractive index of the medium. The refractive index, in turn, affects how light paths are bent through the medium. Ray tracing non-linear media remains a difficult problem to solve efficiently. Furthermore, participating media cause many volumetric light transport effects such as the multiple scattering of light. Multiple scattering is difficult as naively increasing the number of scattered rays through the medium will increase rendering time exponentially. Alternatively, analytical methods can be used for homogeneous volumes. These however cannot be applied to heterogeneous volumes with spatially varying properties.

Extensive research has gone into two different areas: heterogeneous refractive media and multiple scattering of light. Solutions for rendering heterogeneous refractive media limit themselves to single-scattering [5, 28], while solutions towards multiple scattering limit themselves to non-refractive media [25]. Some research has combined both multiple scattering and non-linear media into one solution [9]. Since then, both independent areas have received attention in terms of efficiency and physically based correctness. This thesis aims to cover recent work to find a more efficient combined solution for rendering non-linear media with multiple scattering over existing methods such as non-linear photon mapping [9]. Improving rendering efficiency of a ray tracer is paramount towards making complete physically based rendering more accessible in the film and gaming industry. Ray tracing has been used in the film industry for decades, but often many effects are discarded or simplified due to rendering cost constraints. This constraint is especially holding back the use of ray tracing in interactive applications where frames must be rendered many times per second.

3.1 Contribution

With this thesis the following contributions are made:

- A more efficient framework for rendering multiple scattering in non-linear media than current state of the art.
- A scalable Virtual Ray Light (VRL) framework for non-linear media using lightcuts.
- Stochastic lightcuts for VRLs in a non-linear media.

4 Motivation

The primary focus of this thesis surrounds the following question: How can heterogeneous refractive media with multiple scattering of light be rendered more efficiently than current state of the art?

Rendering in a physically based manner revolves around simulating real-world light transport which is a challenging task that is still being actively researched today. In an ideal scenario, a ray tracer would render phenomena that are a logical result of the laws of physics that govern the real world [9, 23]. A particularly interesting source of natural phenomena are caused by participating media with a spatially varying refractive index. Two examples can be observed in Figure 1. The first shows a superior mirage over a ship on open waters. This is caused by spatial variations in temperature and air density causing light to bend downwards which causes a reflection of the boat above it. The second example shows a science experiment with a laser shining into a volume of water with gradient density causing the laser to be bent. This research area has already seen a lot of progress [9, 2, 14, 5]. While doing an initial survey it was found that non-linear ray tracing research limits itself to single-scattering of light. In many participating media such as water, skin, smoke, or clouds this cannot render the full range of volumetric effects. Multiple scattering causes light to spread the incident light distribution in a spatial and angular manner [27]. Furthermore, absorption of light by the medium affects the amount of spread where the more light is absorbed the narrower the spreading of light. The question remains: how can these effects be rendered together in a generic way such that it can support a wide variety of volumetric non-linear effects? And what recent work can we utilize to increase convergence speed and thereby enhance efficiency over the current state of the art [9]? To answer these research questions we will first provide a brief overview of preliminary information necessary to understand prior work (Chapter 5). Then, we provide an extensive literature study providing an overview of prior work related to non-linear media and multiple scattering (Chapter 6). The findings from this literature study is used to formulate more concrete research questions (Chapter 7). To answer these questions an implementation is made (Chapter 8) to provide results to evaluate whether our approach is more efficient (Chapter 9). Finally, the results are discussed (Chapter 10) and the research questions are answered (Chapter 11).

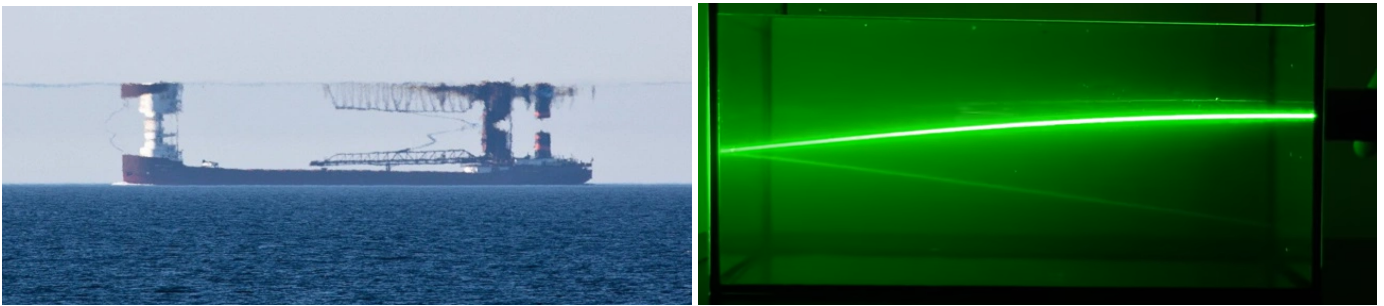


Figure 1: Two examples of non-linear media in the real world. On the left, a superior mirage caused by spatial variations in temperature and pressure, causing a reflection of the distant ship. On the right, a green laser being continuously bent downwards due to a gradient density.

5 Preliminaries

In this chapter the necessary preliminaries will be explained before delving into multiple scattering and non-linear ray tracing methods.

5.1 Participating Media

The following section will explain the building blocks for rendering volumes which is necessary to understand the difficulties of rendering heterogeneous media with multiple scattering.

5.1.1 Radiative Transfer Equation

Ray tracing objects is achieved by shooting rays from the camera into the scene that interact with intersecting surface boundaries. However, within participating media this interaction can happen at any point within the medium, not just at the boundaries. Generally, volumes are modeled as a density per unit volume ρ (m^{-3}). Per unit distance traveled a photon has a probability of interacting with the medium by absorption (μ_a) or scattering (μ_s). The extinction coefficient $\mu_e = \mu_a + \mu_s$ is the probability density of either interaction occurring. The Radiative Transfer Equation (RTE) describes these interactions between a photon and the medium as the difference in radiance along direction ω at point \mathbf{x} [6]:

$$\begin{aligned} (\omega \cdot \nabla)L(\mathbf{x}, \omega) &= -\text{absorption} - \text{outscattering} + \text{inscattering} + \text{emission} \\ &= -\mu_a(\mathbf{x})L(\mathbf{x}, \omega) - \mu_s(\mathbf{x})L(\mathbf{x}, \omega) + \mu_s(\mathbf{x})L_s(\mathbf{x}, \omega) + \mu_a(\mathbf{x})L_e(\mathbf{x}, \omega) \end{aligned} \quad (1)$$

Each term models one event. Radiance is lost due to absorption and out-scattering and gained due to in-scattering and emission. See Figure 4 for a graphical representation. A scatter event can be either elastic or inelastic. Elastic scattering only changes the direction of the ray, while inelastic scattering also shifts the wavelength of light. It is important to note that the coefficients μ_a and μ_s are a function of the point within the medium \mathbf{x} (ie. the coefficients can be spatially varying). In a homogeneous media these coefficients can be replaced by constants.

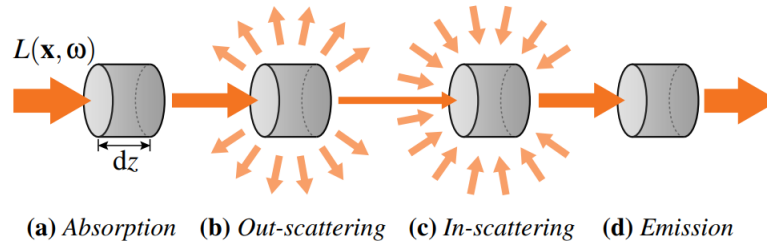


Figure 2: Radiance is lost by absorption (a) and out-scattering (b), and gained by in-scattering (c) and emission (d) [23].

The in-scattered radiance is integrated over the unit sphere S^2 :

$$L_s(\mathbf{x}, \omega) = \int_{S^2} f_p(\omega, \bar{\omega})L_i(\mathbf{x}, \bar{\omega})d\bar{\omega} \quad (2)$$

The RTE 1 is often reformulated by integrating both sides over the direction ω [23]. Here, the gained radiance from emission and in-scattering can be integrated along the input ray $(\mathbf{x}, -\omega)$:

$$L(\mathbf{x}, \omega) = \int_0^\infty T(\mathbf{x}, \mathbf{y}) [\mu_a(\mathbf{y})L_e(\mathbf{y}, \omega) + \mu_s(\mathbf{y})L_s(\mathbf{y}, \omega)] d\mathbf{y} \quad (3)$$

Both the absorption and out-scattering terms are combined into the transmittance function $T(\mathbf{x}, \mathbf{y})$, which is the Beer-Lambert law as a function between two points \mathbf{x} and \mathbf{y} where $\mathbf{y} = \mathbf{x} - y\omega$:

$$T(\mathbf{x}, \mathbf{y}) = e^{-\tau(\mathbf{x}, \mathbf{y})} \quad (4)$$

$$= e^{-\int_0^y \mu_t(\mathbf{x} - s\omega) ds} \quad (5)$$

Where τ is the optical thickness.

$$\tau = \int_0^y \mu_t(\mathbf{x} - s\omega) ds \quad (6)$$

5.1.2 Volume Rendering Equation

The RTE itself only models interactions with volumes, while in reality there are also interactions with objects. Kajiya and Von Herzen [18] introduce the surface rendering equation which was extended to the Volume Rendering Equation (VRE) that models both interactions with volumes and object surfaces:

$$L(\mathbf{x}, \omega) = \int_0^z T(\mathbf{x}, \mathbf{y}) [\mu_a(\mathbf{y})L_e(\mathbf{y}, \omega) + \mu_s(\mathbf{y})L_s(\mathbf{y}, \omega)] d\mathbf{y} + T(\mathbf{x}, \mathbf{z})L(\mathbf{z}, \omega) \quad (7)$$

5.1.3 Phase Function

For objects, the incident light scattered towards another outgoing direction is measured by the bidirectional reflectance distribution function (BRDF). Analogously for volumes, the incident light scattered at a point towards another outgoing direction within the medium is measured by the phase function f_p . The phase function can be either identical throughout the medium, called an isotropic phase function, or spatially varying, called an anisotropic phase function. There are many different phase functions available such as the Henyey-Greenstein or Rayleigh phase functions. The choice between different phase functions primarily depends on the material to be rendered.

5.1.4 Transmittance

In homogeneous media, transmittance $T(\mathbf{x}, \mathbf{y})$ is a constant value for any two arbitrary points. In heterogeneous or inhomogeneous media any material property, such as density or refractive index, can vary spatially and needs to be estimated. Computing the transmittance accurately can become time consuming for high resolution data. One of the simplest ways to estimate transmittance is through ray marching. Given a ray, march along it using a constant step size, and at each stepping point retrieve any necessary local information. The optical thickness τ is accumulated at each step. If the next step would exceed the estimated τ then the path terminates. While intuitive, ray marching presents a biased method that cannot accurately sample the heterogeneous media particularly if it contains thin features.

Regular tracking divides the heterogeneous volume into a set of simpler subvolumes which can be tracked analytically if a closed-form solution for τ exists. Rays are tracked between these boundaries using existing closed-form methods that correspond to that subvolume. This is especially useful for layered volumes such as human skin. While being an unbiased

method and more accurate than ray marching, it brings along a high computational cost for finding all volume boundaries. Delta tracking is a well-known null-collision method which takes a random walk over a given ray until it terminates by Russian Roulette at a collision point based on the local extinction coefficient. Null-collision methods introduce fictitious matter, visualised in Figure 4 by the red particles, to allow closed-form collision sampling methods to be used. When a photon collides with the fictitious matter it simply continues in its original direction.

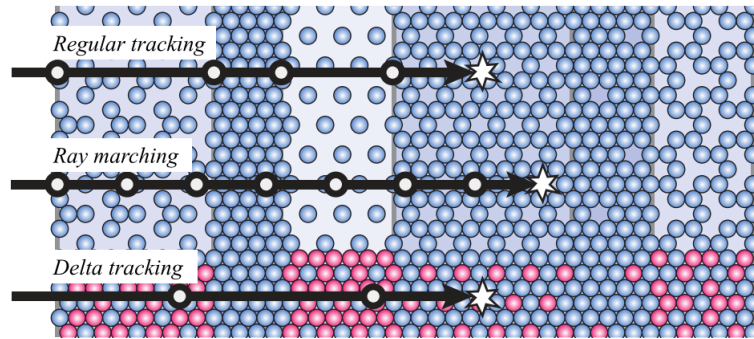


Figure 3: Ray marching, regular, and delta tracking are common approaches towards estimating transmittance [23].

Ratio tracking is a piecewise constant approximation of transmittance [26]. The idea is to calculate a weight based on the product of probabilities from all collision steps until the end of the query ray. This weight is used to scale the contribution of the ray. Although unbiased, this method is especially expensive when there are many steps required to reach the end of the ray.

Residual tracking is a numerical method to find a piecewise exponential solution to transmittance [26]. It divides the extinction coefficient into a control $\mu_c(\mathbf{x})$ and residual $\mu_r(\mathbf{x})$ part. The control part is a coarse approximation of μ_t that is computed using a closed form method. The residual part is the difference $\mu_r(\mathbf{x}) = \mu_t(\mathbf{x}) - \mu_c(\mathbf{x})$ to the exact extinction coefficient which is computed numerically, for example using delta or ratio tracking. This allows residual tracking, in most scenarios, to propagate using greater step sizes than ratio or delta tracking and is therefore more efficient [26].

5.2 Refraction

The following section describes how light refracts which is necessary to understand non-linear ray tracing methods where light is refracted continuously throughout the media.

5.2.1 Law of refraction

The index of refraction is defined as $n_i = \frac{v_i}{c}$, where v_i is the speed of light in medium i and c is the speed of light in a vacuum. Snell's law describes how the light path bends when a photon transitions from one homogeneous media with index of refraction n_1 into another homogeneous media with index n_2 .

$$n_1 \sin \theta_1 = n_2 \sin \theta_2 \tag{8}$$

Here θ_1 is the angle between the interface normal and the incident ray, and θ_2 is the angle between the same interface normal and refracted ray direction.

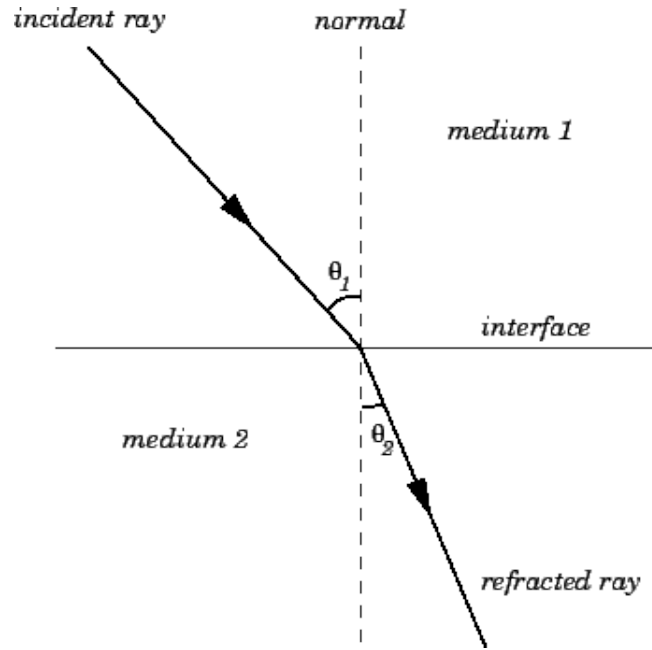


Figure 4: Law of refraction.

Snell's law can be derived from Fermat's principle which is referred to as the second law of refraction. Gutierrez et al. [10] define Fermat's principle as "light, in going between two points, traverses the route l having the smallest optical path length L ". The optical path length $L = nl$, where n is the refractive index and l the traveled route. Fermat's law states that the optical path must be the smallest possible path. The first law of refraction states that the incident ray, refracted ray, and boundary normal all exist in the same plane.

5.2.2 Refractive Radiative Transfer Equation

Ament et al. [2] extend the RTE to the Refractive Radiative Transfer Equation (RRTE) which takes into account global illumination in non-linear media. Their contribution is unique as it guarantees the conservation of energy by replacing radiance with basic radiance. Basic radiance \tilde{L} is a different quantity of radiance which is defined as $\tilde{L} = \frac{L}{n^2}$. Previously in the computer graphics field, the theory of basic radiance has been employed to scale radiance at the interface between two homogeneous media with a different index of refraction [29]. When light enters another medium with a different index of refraction the flux density changes for the refracted ray. Veach [29] shows that this can be solved for discontinuous refraction by scaling the radiance:

$$L_2 = \frac{n_2^2}{n_1^2} L_1 \quad (9)$$

However, Veach [29] employs a nonsymmetric Bidirectional Transmission Distribution Function (BTDF) which is unsuitable for continuously refracting media [2]. The RRTE generalizes the approach by Veach [29] towards continuously

refracting media. Furthermore, whereas the RTE assumes an infinite speed of light the RRTE takes into account the variations of the finite speed of light within the medium. The RRTE defines radiative transport by:

$$\frac{d\tilde{L}}{ds} = \sigma_a \tilde{L}_e - \sigma_t \tilde{L} + \frac{\sigma_s}{4\pi} \int_{\Omega} P(\omega', \omega) \tilde{L} d\omega' \quad (10)$$

where

- $\tilde{L} = \tilde{L}(\mathbf{x}, \mathbf{l}, v, t)$ is basic radiance.
- $\tilde{L}_e = \tilde{L}_e(\mathbf{x}, \mathbf{l}, v, t)$ is emissive basic radiance.
- $n = n(\mathbf{x}, v)$ is the varying refractive index based on position \mathbf{x} and frequency v .
- $\sigma_a = \sigma_a(\mathbf{x}, v, t)$, $\sigma_s = \sigma_s(\mathbf{x}, v, t)$, and σ_t are the spatially varying and frequency dependent absorption, scattering, and extinction coefficients respectively.
- $P(\omega', \omega)$ is the phase function.

Previous non-linear methods based on the RTE ignore the conservation of energy in continuously refracting media [9, 5, 14]. With the introduction of the RRTE it became clear that a good amount of radiance was lost within the non-linear media and that the RRTE is a required extension of the RTE for rendering physically based non-linear media. For their experiments photon mapping was used as it handles single-scattered light in a simple manner. In linear media, straight shadow rays can be traced to sample each light source, so photons are only stored after at least 2 bounces. However, in continuously refractive media tracing a curved shadow ray is not possible as there exists no solution to finding a curved path connecting two points [2]. Therefore, photon mapping is used to store both direct and indirect illumination, therefore providing a physically based solution without tracing shadow rays. However, this increases the number of required amount of photons and significantly decreases convergence speed.

6 Literature Study

This section covers previous work in the fields of participating media, particularly related to non-linear media and multiple scattering.

6.1 Multiple scattering

This section provides a brief overview of work related to efficient ray tracing of multiple scattering within participating media. For a more comprehensive overview of volumetric light transport methods we refer the reader to the extensive survey by Novak et al. [23].

Accurate simulation of light within participating media includes multiple scattering. It is challenging to render multiple scattering efficiently as it significantly increases the number of rays to be traced by a Monte Carlo estimator. Alternatively, analytical methods can be used which are strictly limited to homogeneous media.

In this thesis, we are interested in media with a spatially varying refractive index and thus heterogeneous media. General participating media with the effect of multiple scattering can be rendered using many-light approaches. Many-light methods are a subset of bidirectional path tracing where photons are propagated from both the camera and light sources. Many-light methods are appealing as they provide an adaptive framework for global illumination where it is convenient to trade convergence speed for some consequent bias [7].

6.1.1 Virtual Point Lights

Instant radiosity is a two-pass bidirectional approach [19]. The first pass propagates flux from light sources into the scene. Wherever this flux interacts with diffuse surfaces a Virtual Point Light (VPL) is stored. The second pass traces paths from the camera and estimates in-scattered radiance from the VPLs using standard direct illumination techniques.

The contribution of one VPL is defined as:

$$L_i \approx \Phi \frac{f_s(\theta_p) f_s(\theta_u) T_r(w) V}{w^2} \quad (11)$$

where

- Φ is the flux of the VPL.
- $f_s(\theta_p)$ and $f_s(\theta_u)$ is the phase function for the scattering angle θ_p and θ_u respectively.
- $T_r(w)$ is the transmittance between the query point and the VPL.
- V is a binary term for the visibility between the query point and the VPL.
- w is the distance between the query point and the VPL.

6.1.2 Virtual Ray Lights

The virtual ray lights (VRL) method [25] extends the VPL approach by storing entire ray segments propagated from light sources to efficiently render participating media with multiple scattering in an unbiased manner [25]. Figure 5 shows a visual representation of the difference between the VPL and VRL method. The primary benefit of the VRL approach over VPL is that it converges much faster while requiring fewer virtual lights. Each virtual ray light acts as an infinite set of VPLs which is pruned based on the viewing ray sampling the radiance within the medium. Furthermore, more densely

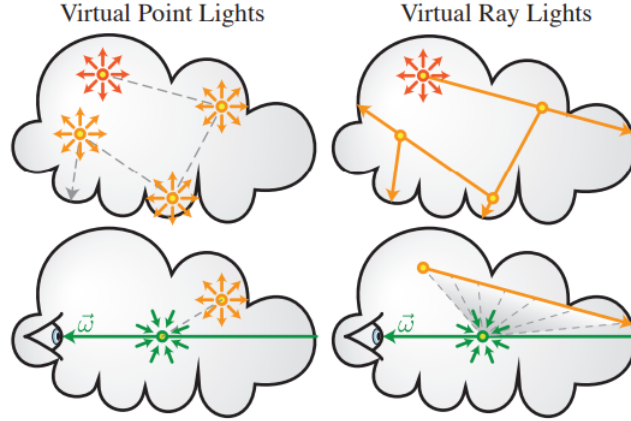


Figure 5: Within a medium, VPL methods store the positions and associated radiance anytime a photon scatters during its propagation from a light source. VRL instead stores the entire ray and its associated radiance [25].

sampling radiance within a media also allows the VRL method to produce results faster without distracting artefacts that are prevalent with VPL based methods. Even with a low number of VRLs the artefacts are much less pronounced.

The contribution of a VRL extends the contribution equation from a VPL 11. For any arbitrary camera path segment and VRL the contribution is found by solving a double-integral:

$$L_m \approx \Phi \int_0^s \int_0^t \frac{\sigma_s(u)\sigma_s(v)f_s(\omega_u)f_s(\omega_v)f_r T_r(u)T_r(v)T_r(w)V(u,v)}{w(u,v)^2} dv du \quad (12)$$

where

- u is a point on the camera ray and v a point on the VRL.
- σ_s is the scattering coefficient.
- t is the length of the VRL denoted by v
- $V(u,v)$ computes binary visibility between u and v .
- $w(u,v)$ computes distance between u and v .
- f_s is the phase function, used twice for both the camera ray and VRL.

When estimating the contribution of a VRL onto a point on the surface of an object this equation reduces to a one dimensional problem.

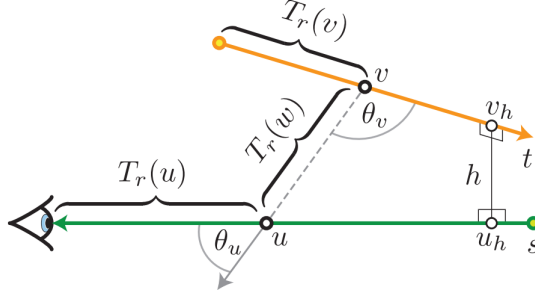


Figure 6: Illustration of Equation 12. Image retrieved from [25].

Similar to the Rendering Equation [18], there does not exist a closed-form solution for Equation 12. Thus, the integral is evaluated using Monte Carlo integration. Let Equation 12 be denoted as $g(u, v)$, then the Monte Carlo estimator is defined as:

$$L_m(\mathbf{x}, \vec{\omega}) \approx \frac{1}{N} \sum_{i=1}^N \frac{g(u_i, v_i)}{pdf(u_i, v_i)} \quad (13)$$

The pdf needs to model g as close as possible such that the integral can be estimated efficiently.

6.1.3 Virtual Beam Lights

While VRLs effectively reduce the number of necessary virtual lights over VPLs there are still distracting singularities visible in the early stages of rendering. Various clamping methods have been proposed for VRLs which eliminate the singularities but also introduce bias. Novák et al. extend upon their own work with Virtual Beam Lights (VBL) which are inflated VRLs allowing for faster convergence at the expensive of introducing bias [24]. Inflating the VRL into a VBL is achieved, conceptually, by inflating all points on the VRL into spheres, thereby creating a volumetric beam and effectively blurring the radiance. This blurring of a VRL is what allows VBLs to eliminate the distracting singularities much faster. The main advantage of VBLs over clamped VRLs is possibility to take advantage of progressive radius reduction. The authors state that this allows the VBL algorithm to convergence to an image without bias and variance in the limit. This allows both fast previews and unbiased results to be obtained using the same algorithm. Figure 7 shows how the early stages of rendering are clearly biased albeit without distracting artefacts and that overtime the image converges closer to the reference.

6.1.4 Many-light Optimizations

Lightcuts When tracing camera rays through the scene the VPLs can be selected using importance sampling. However, this results in rendering time growing linearly as the number of VPLs increase. Extensive research has gone into optimizing this many-lights problem. One of such optimizations is Lightcuts [32] which hierarchically clusters VPLs into a tree such that the illumination from many lights can be retrieved in sublinear time. At the leaves are the individual lights, at each parent a cluster containing all the children lights, and at the root the cluster representing all lights in the scene. The direct illumination from the lights in a cluster is evaluated using a representative light which is similar in its material, geometry, and visibility to the other lights in the cluster. The light tree is precomputed before rendering. While rendering, the light tree is used to find a partitioning of the lights that is appropriate for the input point. Some pixels may use one

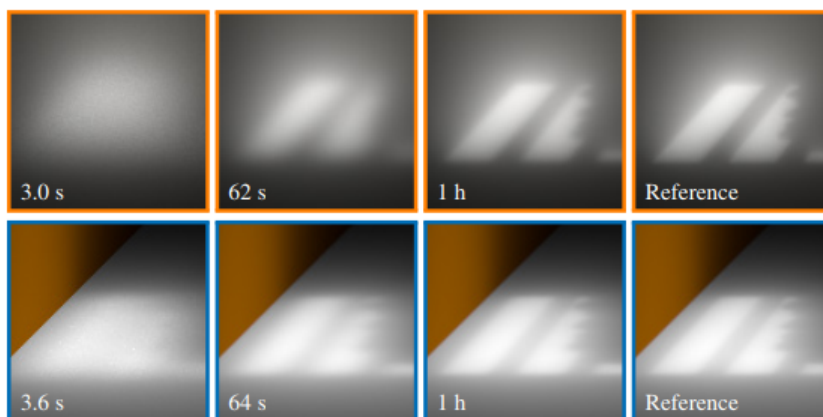


Figure 7: VBLs using progressive radius reduction convergence to the reference image.

cluster while other pixels use the individual lights in that cluster for increased accuracy. There is always a certain error to the final image when choosing to use a cluster instead of individual lights, but this is not noticeable if the error stays below a perceptual visibility threshold. This threshold is a user-specified parameter.

VRL Lightcuts More recently, Vibert et al. [30] applied lightcuts to VRLs. The light tree itself is relatively similar, except that the clusters now contain VRLs instead of VPLs. Retrieving the illumination contribution between light path segments involves a more complex Monte Carlo estimator than for querying points. Thus, a new upper bound on the maximum contribution of a VRL cluster is derived. The lightcut scalability is highly dependant on the tightness of this bound such that as many lights as possible can be cut without introducing artefacts. The challenge with clustering VRLs is that bounding boxes around VRLs are much more likely to overlap with one another than VPLs. In its current stage, this gives lightcuts for VRLs a lower scalability than for VPLs, but it still provides more than one order of magnitude speedup over the original method. The authors limit themselves to homogeneous media by assuming a transmittance of 1 throughout the media. Further research is required to find an efficient method for querying the transmittance in heterogeneous media. This is not necessarily a trivial task as estimating transmittance can be time consuming. A looser upper bound on the light tree may be acceptable in such cases.

Stochastic Lightcuts One problem with lightcuts is that it is temporally unstable due to sampling correlations that are the result of using the same light tree for the entire image. As light tree construction is expensive, both in computation time and memory size, it is not practical to create multiple light trees. Cem Yuksel [34] modifies the lightcuts method by implementing stochastic sampling. When evaluating the light tree, the representative light source of a cluster is ignored in favour of a random light source. The illumination from a cluster can then be estimated by scaling the illumination of the random light with the probability of choosing it. Thus, stochastic lightcuts have a randomized instead of predefined order of evaluating lights which removes the sampling correlation. Furthermore, as it does not rely on representative lights it does not matter what type of light is used in the tree, potentially making it a feasible solution for VRLs as well.

Lin et al. [21] extend stochastic lightcuts to work on the GPU, thereby making the method suitable for real-time applications. They propose to make this possible using a perfect binary light tree, which is extremely fast to construct and traverse, and cut sharing between neighbouring pixels. Both solutions help to minimize construction and light sampling time.

Matrix Row-Column Sampling Another alternative to solving the many-lights problem is by matrix row-column sampling (MRCS) [13]. Conceptually, the many-light problem is viewed as a large matrix of interactions between lights and samples. The columns represent all samples lit by one particular light while the rows represent one particular sample lit by all lights. The solution of this matrix can be computed by summing up the columns which results in the contribution of all lights for all pixels. The primary benefit of using this matrix representation of the many-lights problem is that it is especially effective on the GPU as the columns and rows are computed using shadow mapping.

6.1.5 Volumetric Photon Mapping

Volumetric Photon Mapping (VPM) is an extension of photon mapping to support scenes with volumes [17, 16]. VPM work similarly to VPLs for participating media, except that in the first pass the radiance at each intersection of the photon with the scene is stored in a photon map. During the second pass, gathering rays are shot into the scene. Wherever these rays interact with the scene the photon map is sampled for nearby photons. These nearby photons are used to estimate the indirect illumination at the intersection. The key difference between VPM and VPL is where paths connect. With VPM, the gathering ray connects at the position of the photon while VPL always connects to the light source with at least two bounces. The quality of indirect lighting for both VPL and VPM methods is directly dependant on the number of photons traced during the first pass, with too few photons resulting in artefacts that make the virtual lights visible in the final image. Extensive research has gone into reducing the intensity of such artefacts, for example by blurring or clamping the energy [12]. However, such approaches often introduce bias which may be acceptable depending on the use case.

6.1.6 Photon Beams

Jarosz et al. [16] first extend volumetric photon mapping by introducing a beam gathering technique which efficiently gathers all photons along a query ray instead of performing multiple costly point queries. Density estimation is performed along the entire photon path which makes this extension analogous to extending VPLs to VRLs. It shares the same benefit of increased convergence speed as fewer photons need to be traced in order to achieve noise free results. However, the convergence speed of photon beams is still significantly slower in producing artefact free images than VRLs [25]. Photon beams work best for volumetric caustics since those paths are concentrated, but multiple scattering causes an incoherent light path and therefore requires millions of photon beams for artefact free results. Photon beams were later extended to progressive photon beams (PPB) to provide guarantees on convergence speed [15].

6.1.7 ReSTIR

Spatiotemporal reservoir resampling (ReSTIR) is a direct illumination sampling method for scenes with many lights in real time [4]. It builds upon resampled importance sampling (RIS), which is a technique to create an importance sampling estimator for an integral using a set of weighted samples generated from a sub-optimal PDF. ReSTIR utilizes reservoirs, a small fixed-size data structure, for each pixel that store the accepted samples from the re-sampled distribution. These reservoirs allow spatiotemporal reuse of statistics that progressively improve the sampling PDF for each pixel on the screen. ReSTIR is unbiased as only the sampling probabilities are reused, although it can be adapted into a biased version for increased convergence speed.

Recently, ReSTIR was extended to solve the volumetric path tracing rendering equation to enable real-time rendering of heterogeneous participating media with many lights [20]. The VRE 7 is generalized to a path integral representation as the entire paths need to be sampled instead of just light positions. This allows spatiotemporal reuse of path vertices and directions. Path reuse inherently allows reusing transmittance values and thereby avoiding many costly estimations of the transmittance. Once every pixel reservoir has selected a path, RIS is used to combine neighboring reservoirs thereby increasing the number of samples per pixel at a fraction of the cost. Finally, temporal reuse further increases sample quality by temporally reprojecting samples from prior frames. The problem with temporal reuse for volumes however is that there

does not exist a correct motion vector for an animated media. This is solved probabilistically by finding motion vectors based on motion vectors of the mediums in a pixel that are likely to select useful reservoirs of prior frames. Transmittance estimation for importance sampling is further optimized by ray marching a coarser piece-wise constant representation of the volume which greatly reduces cost albeit at introducing some noise.

6.2 Non-linear Ray Tracing

It is reasonable to assume that in most environments photons follow straight, linear paths. However, nature includes many phenomena where this is not the case and photons follow curved, non-linear paths. Existing research often focuses either on a general method to support most non-linear phenomena or solutions for specific effects such as heat shimmering, gravitational lensing, and the green flash.

6.2.1 General Methods

Non-linear Photon Mapping Non-linear photon mapping (NPM) extends VPM to provide a full global illumination solution for non linear media [9]. The extension includes solving Fermat’s principle to obtain the curved trajectory of each photon. Shadow rays cannot be used to determine direct illumination at an arbitrary point within the media due to the fact that there does not exist a solution for finding a curved path between two points. It is possible to use shadow rays just like a normal ray tracer would, but this would not yield physically based results. Thus, the photon map stores both direct and indirect illumination to overcome this problem. The primary disadvantage of this technique is its performance, as solving curved path trajectories for photon and view-rays using Fermat’s law is expensive. Photon mapping in general requires a high number of photons to be traced in order to obtain noise-free results. Photon mapping introduces bias as the radiance at a point within the volume is a local average, but the authors do not discuss or evaluate bias.

Eikonal Rendering Ihrke et al. [14] introduce a real-time method for rendering heterogeneous media with a spatially varying index of refraction and anisotropic single scattering based on differential equations derived from the Eikonal equation. Camera rays follow curved paths similarly to NPM [9]. Their method is unique as the irradiance distribution is estimated using adaptive wavefront tracing. Given a heterogeneous media with n refractive indices, the motion of a photon can be derived from the Eikonal equation as $\frac{d}{ds}(n \frac{dx}{ds}) = n$.

Their method ignores multiple scattering and is limited to spatially confined objects. Light sources are formulated as a vector field of directions towards local lights along with a scalar field with differential irradiance values. These are computed using adaptive wavefront tracing propagated from the light sources.

Interactive relighting of dynamic refractive objects Sun et al. [28] present a novel technique for rendering refractive objects that allow for adjustments to the physical parameters of the medium at runtime. The method uses adaptive non-linear photon tracing, which advances photons from a light source along their curved paths while leaving radiance in a volumetric texture. An octree is used to determine the largest step size possible for the photon to traverse with a near constant index of refraction. The volume of radiance distributions filled by these photons is smoothed using a Gaussian kernel to reduce noise. After the photon tracing pass, the image can be rendered using viewing rays from the camera. Viewing rays that intersect with the medium are marched through the medium with a fixed step size. Most non-linear media in the physical world have a smooth refractive index gradient allowing piece-wise methods to take reasonably large steps, each assuming a constant refractive index. However, if the refractive index gradient has a high magnitude then a large number of tracing steps are required to achieve reasonable results, making the method unsuitable for interactive applications for certain media.

Interactive rendering of non-constant refractive media Cao et al. [5] instead define a piecewise-linear function on a tetrahedral mesh to model the varying refractive index in a media. The benefit of using a tetrahedralization over a structured grid is that it can more accurately model unstructured meshes and trace analytic ray curves efficiently [5, 22]. Unstructured meshes are advantageous as they can adapt to many different domains, such as atmospheric flow simulations. All vertices of the tetrahedralization have a constant index of refraction. When sampling the volume, the tetrahedron that contains a query point can be retrieved efficiently. The refractive index within the tetrahedron is the linear interpolation of the 4 constant refractive indices associated with the 4 vertices of that tetrahedron. However, their method is limited to single elastic scattering.

Mo et al. [22] extend upon the work of [5] by making their method suitable for both curved light and sound wave propagation, larger scenes, and improve upon the efficiency of the curved ray traversal algorithm. Given a set of uniformly distributed points describing some media profile, the set is resampled according to the magnitude of refractive index variations within the media. This resampled point set is transformed into a tetrahedralization to efficiently traverse the media. Each tetrahedron estimates the local gradient of the participating media which the light rays use to follow curved paths. However, the method does not handle single or multiple (inelastic) scattering and absorption and compared to piecewise-linear methods the curved ray traversal does not seem to have a significant benefit in the quality of the final image. Curved ray traversal is advantageous in media with a high magnitude in the refractive gradient greatly reducing the number of traversal steps.

6.2.2 Atmospheric Phenomena

The density of a participating medium affects the speed of light [3]. Often, density is spatially varying due to, for example, spatial variations in temperature. Thus, where temperature changes rapidly the light paths bend the most considerably towards the cooler air. A *superior* mirage (see Figure 1) occurs when temperature increases with height, whereas an *inferior* mirage occurs when temperature decreases with height.

Zhao et al. [35] introduce a physically-based framework for rendering thermal flow simulations, particularly the effects of heat shimmering and mirages. Heat shimmering is also caused by spatially varying temperatures that affect the index of refraction and thus how light bends through the medium. Non-linear ray tracing is achieved by traversing the volume in small step sizes. At each step the temperature field gradient is computed which can be translated into a refractive index by any arbitrary function that models the material, for example a refractive index function for air could depend on temperature, moisture, and pressure. The bending of light is simply computed using Snell's law. This continuous bending may cause light paths to bend so far that it reflects, also called *total reflection*. This is why, for example, a reflection in the inferior or superior mirage can be seen.

Gutierrez et al. [11] present a general physically-based framework for simulating many atmospheric phenomena taking into account heterogeneous media. However, their method assumes that the heterogeneous media can be separated into multiple homogeneous media such that the boundary between these media can be used to bend the light paths using Snell's law. This is a reasonable assumption for media with a low gradient in its refractive index field, such as the atmosphere.

6.2.3 Gravitational lensing

Groller [8] presents a variety of methods for rendering non-linear phenomena surrounding various types of non-linear rays. The first type of non-linear ray are those caused by strong gravitational sources. The gravitational effect is attenuated by the distance between the ray and the gravitational source. Any number of gravitational sources can be used by summing up the gravitational forces towards each sources attenuated over the distance squared. The second type of non-linear ray are defined by chaotic dynamic systems commonly specified by an attractor such as the Lorenz or Rossler attractors. Finally, a more general way for representing non-linear rays are using parametric functions.

7 Research Methodology

7.1 Research Aims

With a better understanding of the non-linear ray tracing and multiple scattering computer graphics research areas we are able to more concretely define the research question into several aims.

1. *Q1: To explore how light paths can be reused for efficient ray tracing of non-linear media with multiple scattering.* Many-light rendering methods such as photon mapping have already proven useful for non-linear media with multiple scattering [9, 2]. Analogous to photon mapping are virtual point lights (VPLs) which have been extended toward virtual ray lights (VRLs) which significantly reduce the number of required photons to propagate from a light source to simulate multiple scattering in a participating media. Furthermore, lightcuts can be applied to VRLs making it a scalable solution. Theoretically, nothing prevents VRLs from being used in a non-linear environment albeit in a biased manner. How could VRLs be used in a non-linear environment and what is the bias that they introduce?
2. *Q2: To figure out how VRLs with lightcuts can be used for heterogeneous refractive media.* Lightcuts have already been applied to VRLs in a homogeneous media which enhances scalability tremendously [30]. Theoretically, it is possible to extend this method towards heterogeneous refractive media. The authors suggest a transmittance data structure for quickly sampling an estimate of the local transmittance. For heterogeneous refractive media the extinction and scattering coefficient can remain spatially constant. Thus, medium scattering will occur homogeneously and thus there should be no need for the transmittance data structure.
3. *Q3: To determine whether stochastic lightcuts can be applied to virtual ray lights.* Theoretically, it is possible to apply stochastic lightcuts to a light tree containing any light source of arbitrary shape. Stochastic lightcuts enhances convergence speed by removing the predefined order in light tree refinement which allows early stopping [33]. Could stochastic lightcuts allow interactive previews when combined with VRL lightcuts?

It is important to note that using VRLs as they are proposed [25] in a non-linear environment may not be physically correct. This holds true for any VPL based framework. Photon and camera rays can be propagated over non-linear paths, but integrating the radiance on a camera ray from a VRL assumes a linear environment. However, it is expected that this will not affect the final image in a noticeable way particularly in media with a smooth refractive index gradient. To our knowledge, VRLs have not yet been used in a non-linear environment.

7.2 Method

As a first step, we will establish the ground truth by implementing non-linear volumetric photon mapping (VPM) [9], excluding inelastic scattering. Thus, non-linear photon mapping is simply photon mapping for both direct and indirect illumination where photons propagate along curved paths. Next, the VRL method is implemented as proposed by the original paper [25]. This implementation will be compared to the established ground truth to ensure the correctness of the implementation before extending it to handle efficient lightcuts for heterogeneous refractive media. The final image should be visibly identical to VRLs without lightcuts. The scalability with the number of VRLs will be evaluated. The VRL lightcut method is then extended even further by utilizing stochastic lightcuts. As VRLs only compute indirect illumination the VRL method must be combined with VPM for direct and caustic illumination. Finally, the VRLs are extended to follow curved instead of linear trajectories. The simplest implementation for non-linear ray tracing is to take random small steps, evaluate the local refractive index, and continuously curve the path accordingly. Then, the VRLs can be used in a non-linear environment and their results can be compared to non-linear photon mapping in terms of Root Mean Squared Error (RMSE), convergence speed, and computational resource cost.

8 Implementation

8.1 Integrators

8.1.1 Volumetric Photon Mapping

The complete estimate of the volumetric rendering equation is computed using three separate photon maps: global, caustic, and volume. During the first pass, the pre-processing stage, photons are propagated from the light source(s) and traced through the scene. A photon may interact with the scene geometry or volumes. Depending on each interaction the photon will be stored in one of three photon maps: global, caustic, and volume. If the photon interacts with a diffuse surface it is stored in the global map. If the photon came was bounced from a specular material and interacts with a diffuse surface it is stored in the caustic map. Finally, if a photon undergoes a scattering event within a participating medium it is stored in the volume map. In contrast to traditional photon mapping, if the first interaction of a photon is a scattering event within a medium it is stored in the volume map as well. These photons represent direct illumination. Then, during the second pass we trace camera rays. If the camera ray intersects with a medium the photons are gathered along the entire ray segment that is the intersection between the camera ray and medium bounding box. The implementation includes a target number of photons per map. This allows experiments where, for example, only the number of volume photons change.

8.1.2 VRL

From a top-down perspective, the VRL implementation follows a similar structure to VPM. The first pass traces photons from light sources, but instead of storing just the vertices, the entire path segment is stored as a virtual ray light. Only direct and caustic photons are stored in a volume photon map as these effects cannot be rendered using VRLs. During the second pass, primary rays are cast from the camera into the scene. The intersection of this ray with any potential medium is used to query the contribution of VRLs.

Computing the contribution of all VRLs for a given camera ray is the sum of all individual VRL contributions. $T_r(u)$ and $T_r(v)$ are trivially computed while $T_r(w)$ also includes the visibility term $V(u, v)$. If there is any surface with a non-null BSDF (a BSDF that changes the direction of the ray) between u and v then $T_r(w)$ returns 0.

Choosing u and v can be done through uniform or importance sampling. Uniform sampling produces noise while importance sampling effectively removes this noise in just 1 sample with almost no overhead (see Figure 8). Importance sampling is implemented as proposed by Novak et al. [25]. The integral includes many terms which are reflected in the PDF, such as scattering coefficients and phase function.

8.1.3 VRL Lightcuts

The lightcut is constructed using agglomerative clustering [32, 30]. Each VRL is inserted into a leaf node which determines its bounding box (AABB) and added to a queue. Then for each leaf node, the algorithm finds another leaf node in the queue that is closest. This distance is measured by the sum of flux multiplied by the extent of the minimum AABB that contains both nodes. If neither of the two nodes has a parent, then a parent node is made for them. The node pair is removed from the queue and the new parent node is added to the queue. The representative light is chosen randomly between the representative lights of its children. This process is repeated until only one node remains in the queue which will be the root node.

Querying the lightcut starts at the root node which is continuously cut until the error is below the upper bound.

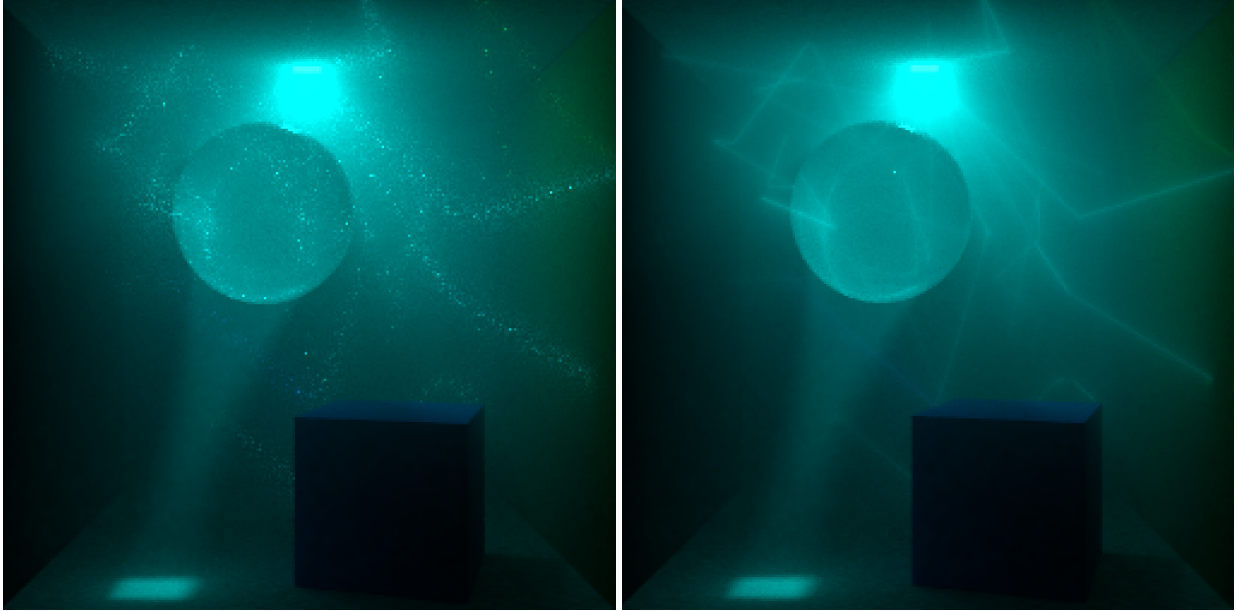


Figure 8: VRL uniform sampling (left) compared to importance sampling (right) with 100 VRLs in a linear homogeneous media.

8.1.4 Stochastic Lightcuts

Stochastic lightcut (SLC) randomly chooses a new representative light at every node evaluation step. This selection process is guided using importance sampling between the representative lights of the two children. The simplest is according to the intensity I of the light i . For a VRL this is simply the flux multiplied by its length. By removing the sampling correlation, stochastic lightcuts allow early stopping of light tree refinement after one or more leaf node evaluations. The implementation includes a parameter for this value. More evaluations reduce noise but negatively impact render time. In order to work with multiple threads the entire lightcut object is copied for each available thread. This is necessary as the random representative lights need to be temporarily stored somewhere. If threads were to access the same lightcut then they will overwrite the representative lights for other query rays which produces significant artefacts.

8.1.5 Non-linear Media

While there are many efficient methods available for tracing curved rays, for example using an octree [28] or tetrahedralization [5], it does not matter for the aims of this research how efficient it is. Thus, we opt for the most implementation friendly approach. Support for non-linear media was implemented as an extension of the homogeneous media such that the only heterogeneous property of the medium is the index of refraction. The bounding box of the medium is discretized into a three-dimensional grid where each cell has a constant index of refraction. The refraction indices can be retrieved from a volume data file or computed using a function, for example using linear interpolation or the Gladstone-Dale law [9].

Whenever a ray interacts with the bounding box of a medium the tracer samples a medium interaction along the ray. If the medium is non-linear, then additional non-linear interactions (NLIs) are sampled. Any NLIs that happen before the MI are handled first. A NLI represents the intersection of the ray with the index of refraction grid. The NLI is identical to an

interaction on a surface with a dielectric BSDF. The ray is refracted using Snell's law. The medium interaction is updated as the position is changed due to the refracted ray. For the preprocessing stage, a VRL is stored from the ray origin to the position of the NLI if the NLI caused a change in ray direction. In case of high resolution IoR grids, a minimum length for the VRL is set to prevent the VRL from being cut too many times. This would otherwise require a high number of very short VRLs in the lightcut. The resolution of the non-linear media has to be sufficiently high such that total internal reflection does not occur artificially.

9 Results

The non-linear VPM and VRL integrators were implemented in Mitsuba 2 [1]. Both preprocessing and rendering happen on the CPU. All experiments were executed on a desktop computer with an AMD Ryzen 7 5800x CPU at 3.80Ghz with 8-cores and 32GB DDR4 memory. Preprocessing is single-threaded while rendering takes advantage of all available threads (in this case 16 threads). To compare performance between VPM and VRL we only render medium-to-medium transport as this is the only piece of the VRE that is computed differently between the two integrators. All media utilized an isotropic phase function.

Both integrators were implemented such that they could render the complete VRE. Figure 9 shows a volumetric path tracer which, as expected, shows noise in the volume even after 100k *spp*. The VPM and VRL integrators are much more efficient in rendering participating media. For each scene the difference from reference is expressed as RMSE. Comparing the RMSE between the VPM and VRL renders is not useful as the VPM method produces slightly darker images which would affect the error. This loss in illumination is due to its sub optimal gathering method.

The first scene (Figures 9 and 11) is a Cornell box with a homogeneous linear and non-linear media. In the linear scene, a comparison between a volumetric path tracer, VPM, and VRL is made to establish the ground truth. Then, in the non-linear version, the VPM and VRL integrators are compared for equal pre-processing and render time results.

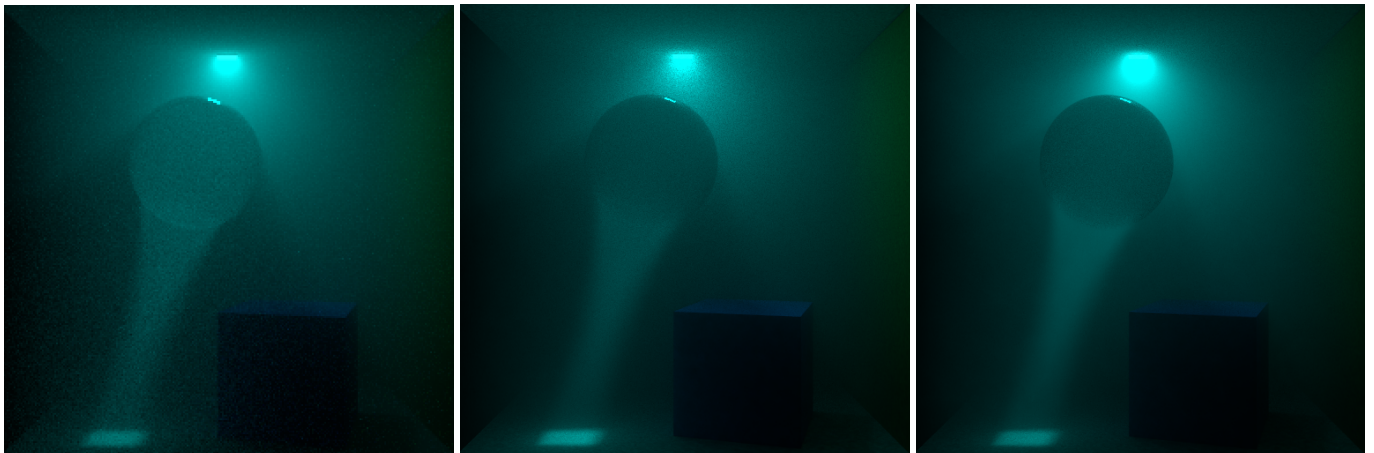


Figure 9: Reference Cornell box renders from a volumetric path tracer (left), VPM (middle), and VRL (right).

The second scene (Figures 12 and 13) is a water tank with a green laser on the right shooting photons into the tank. The water tank has a horizontal mirror at the bottom and a vertical mirror on the left. Here, the scalability of lightcuts in a non-linear media is evaluated and the performance impact of non-linear over linear camera rays is tested. The non-linear media has an IoR of 0.95 at the top and 1.0 at the bottom to cause light to bend downwards. The cells in between the top and bottom have a linearly interpolated IoR. As the IoR only varies in height, the x and z resolution of the grid is 1 while the y resolution is 100. The laser is angled downwards slightly such that the ray intersects with the horizontal planes of the IoR grid.

The third scene (Figures 10 and 16) shows a superior mirage on the horizon with a thin layer of fog. The entire scene is a non-linear volume with an environment map used as the emitter. This is achieved by importance sampling a coordinate on the environment map texture which gives the direction from the center of the scene. The origin of the photon ray is computed by multiplying this direction with the radius of the scene. The direction of the ray is uniformly sampled over

the local hemisphere similarly to area lights. Once again, the IoR only varies in height. The IoR is 1.03 at the bottom and linearly interpolates to 1.0 at the top. This causes light to bend towards the ground causing a superior mirage on the horizon.



Figure 10: Mirage scene without fog rendered with linear rays (top) and curved rays producing a superior mirage (bottom).

9.1 Cornell Box Scene

From Figure 11 we can observe interesting phenomena in the Cornell box scene when changing the medium type from homogeneous to nonlinear. Due to curved camera rays the floor, box, and ceiling appear curved instead of straight. The top of the blue box is more clearly seen and the dielectric sphere is stretched in its height. With 1 spp and a low-resolution non-linear grid, we can also observe several horizontal lines around the horizon. This is a limitation of the non-linear method we use and is resolved using a higher resolution grid and multiple spp. However, this would increase render time to impractical degrees. Due to time constraints, it was decided to accept these slight artifacts as they are unrelated to the VPM and VRL methods. The VPM render also shows aliasing artifacts on the geometry. This is caused by a fixed volume gathering radius and can be resolved by jittering the radius over multiple pixel samples. However, here we only compare 1 spp and therefore this is not possible.

Pre-processing took 15 minutes for both the VPM and VRL methods. This is because both required 5 million surface photons which took longer to trace than the target number of volume photons or VRLs. The 32000 VRLs were gathered within less than 5 seconds and it took an additional 16 seconds to cluster them together for the lightcut. The VPM render uses 5 million volume photons for both direct and indirect illumination. About 20% of photons were direct photons, so 1 million photons were used for the VRL renders. Rendering took 3.5 seconds for VPM and 2 minutes for the VRL LC integrator. An additional 20 seconds was necessary for importance sampling over uniform sampling. Finally, over 2 million queries were made to the VRL lightcut. This is significantly more compared to using linear camera rays which were only 74 thousand. By breaking up the camera ray into piecewise linear segments the number of lightcut queries is increased by a factor of 27 and render time by a factor of 15. This shows that the number of VRL queries grow with an increasing resolution of the non-linear grid.

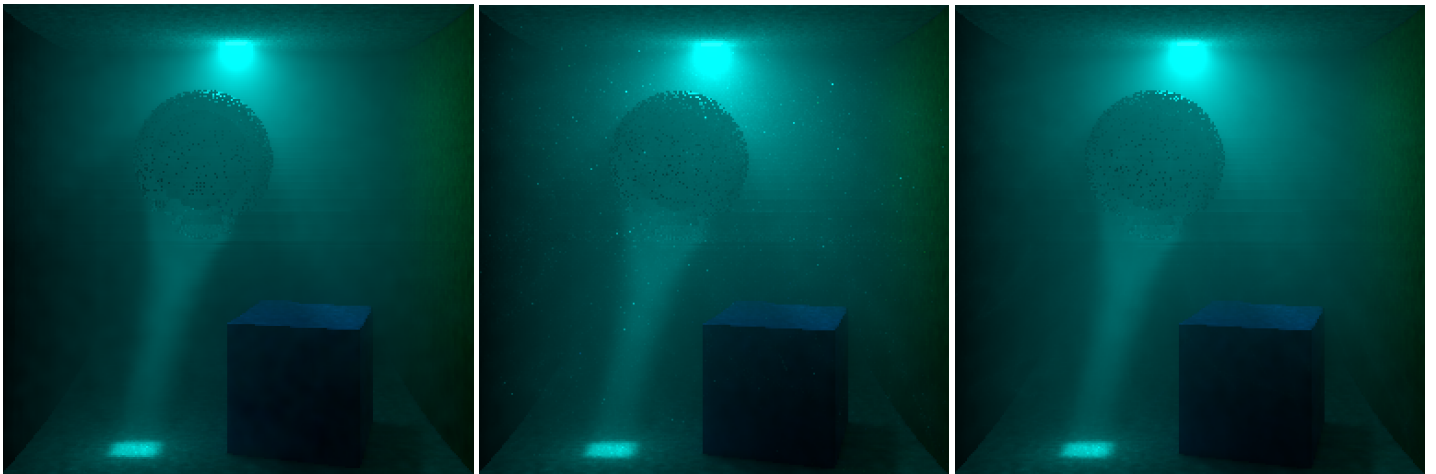


Figure 11: Renders (256x256 with 1 spp) of Cornell box scene with non-linear volume using VPM (left) and VRL LC using uniform sampling (middle) and importance sampling (right).

9.2 Laser Scene

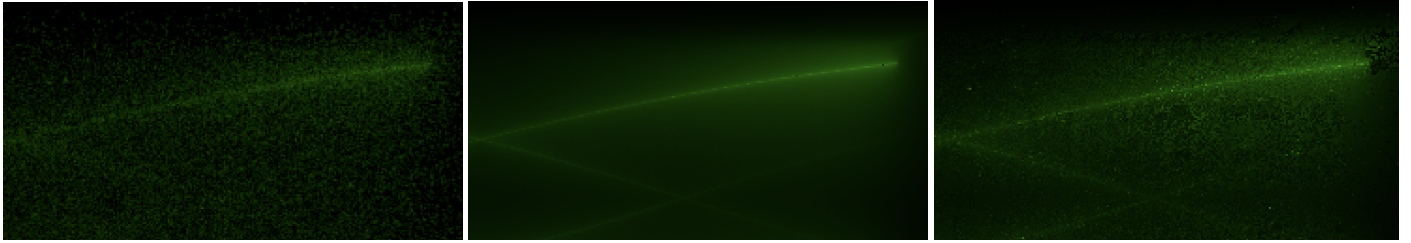


Figure 12: Equal time comparison in non-linear media between VPM (left), VRL LC (middle), VRL SLC (right) using linear camera rays (256x256 with 1 spp).

Linear Camera Rays The laser scene shows an interesting comparison between VPM and VRLs. First, we compare the non-linear media with non-linear photons but with linear camera rays. In equal time at a resolution of 256 by 256 and 1 spp, VPM spends most of its time to sample the medium during the preprocessing stage. It can be observed that 100k photons are not nearly enough to sufficiently sample the medium compared to just 16k VRLs. Especially notice the reflected ray on the bottom mirror which is perfectly visible in the VRL LC and SLC renders but cannot yet be observed from the VPM render. This difference is especially critical in non-linear media as computing curved rays in our case is expensive. The VRL integrator is able to sample the medium with higher quality using fewer emissions and less memory space. The preprocessing stage of the VRL almost exclusively consists of constructing the lightcut. Due to time constraints, this implementation did not feature an efficient nor multi-threaded construction of the lightcut which could greatly speed up this process. Table 1 shows a comparison in computational cost for the equal time renders as shown in Figure 12. As the lightcut is a binary tree and is stored aside from the list of VRLs it takes up 3 times as much storage compared to unoptimized VRLs. The SLC reduces render time by a factor of 3 over the standard LC while adding visible noise. However, the RMSE for VRL SLC is still lower than VPM within less total time.

	VPM	VRL LC	VRL SLC
Photons/VRLs	100k	16k	16k
Volume Map Size	10.7 MiB	4.88 MiB	56.2 MiB
Total Emissions	107900	1130	1130
Pre Time (s)	8.336	4.552	5.251
Pre Trace Time (s)	8.307	0.084	0.09
Map Construct Time (s)	0.029	4.468	5.161
Render Time (s)	0.264	2.717	0.773
Total Time (s)	8.6	7.269	6.024
RMSE	0.22	0.07	0.17

Table 1: Render time and memory breakdown of laser scene as seen in Figure 12.

Non-linear Camera Rays The results look different when we apply non-linear camera rays for the complete effect (see Figure 13). There are a few effects to highlight. The image is only similar to linear camera rays on the horizon. This is where camera rays are shot straight over the horizontal plane where the IoR is constant. Above and below the horizon, the entire laser is moved slightly upwards. The intersections with the left and bottom mirrors also appear slightly higher than where they are. This is especially evident by the mirage occurring near the horizon where the laser appears to be disconnected. This is where the angle of incidence from the ray is below the critical angle, causing a reflection instead of refraction.

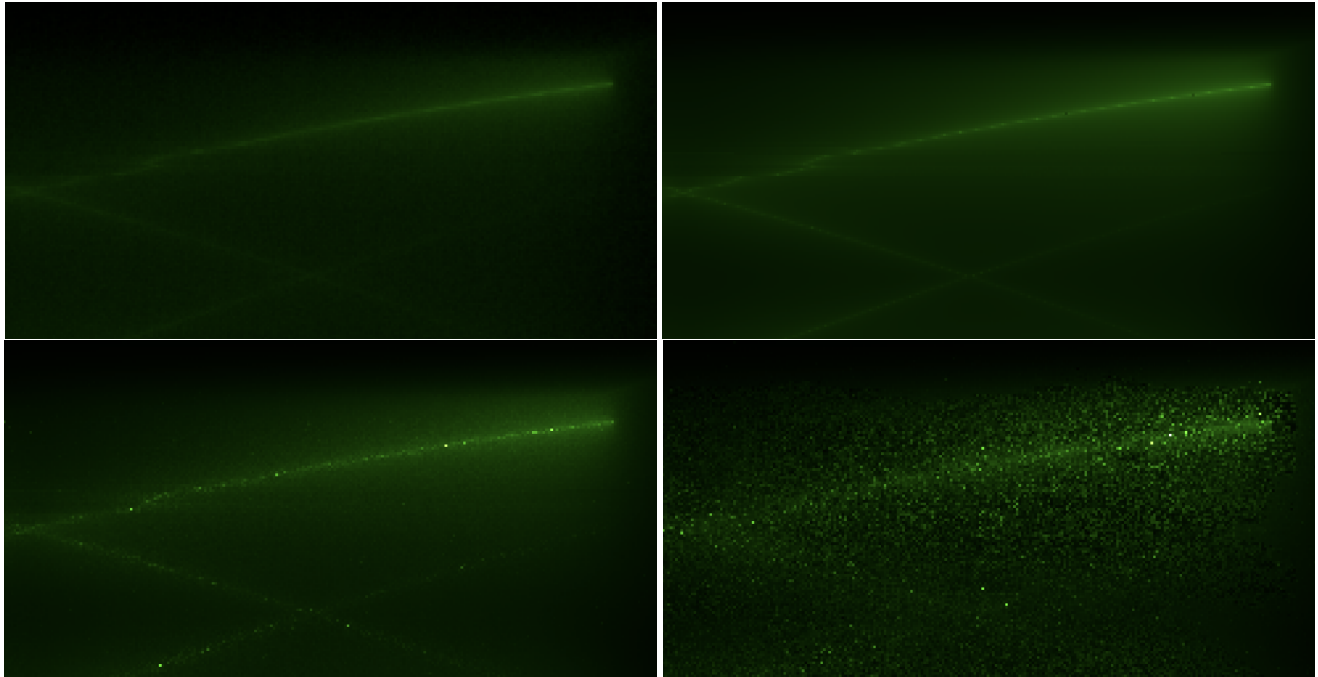


Figure 13: Renders of laser through non-linear media between VPM (top left), VRL LC (top right), and VRL ALC (bottom left) and SALC 10 (bottom right) using curved camera rays (256x256 with 1 spp).

While render time for one spp sees only a slight render time increase for VPM, the render time for VRLs is increased by one order of magnitude. The same 16k VRLs with linear camera rays which took 2.7 seconds to render now take 28 seconds. This is because, for the non-linear camera rays, we require a high-resolution non-linear grid. Every time the camera ray is bent there is a new small camera ray segment that needs to be queried for indirect illumination from the lightcut. This almost is equivalent to taking 10 samples for integrating every VRL to camera ray contribution, depending on the number of times that the ray is bent. As we have seen from the linear camera ray results it is not necessary to take more than 1 sample for each VRL to camera ray integration step. Thus, the VRL integrator was slightly adapted to sample one random point along the entire curved ray. The curved ray is stored as a set of its piecewise-linear ray segments. The results of this adaptation are shown with the VRL atomic lightcut (ALC) and stochastic ALC (SALC). Figure 13 shows that ALC produces a near similar image to VRL LC with 1 spp. As the ALC uses uniform sampling it introduces some noise which is also noticeable in RMSE.

From Figure 14 we can observe the progression of RMSE to the reference. The RMSE is normalized using the maximum RMSE. The VPM reference contained 8 million photons while the VRL reference contained 128k VRLs. The VPM method

needs many millions of photons to converge, while VRL only needs a few thousand samples. Particularly, 8000 VRLs can obtain a lower error than 4 million photons. The total time for the respective VRL render was 25 seconds while the VPM render took 5.5 minutes. This shows a significant reduction in total computation time for a lower error. Furthermore, VRL SALC 1 and 10 mimic the shape of the VRL ALC error curve at a higher error. Here we can see the error even increase slightly with more VRLs in the lightcut. This is due to the stochastic nature of SLC where a higher capacity results in a greater probability of choosing a suboptimal light in the lightcut. The RMSE flattens out after 8000 VRLs for SLC 1 and 10 as more virtual lights will not reduce the inflicted noise, but more pixel samples will. While noise and RMSE are increased compared to ALC, both settings significantly reduce render time per pixel sample. More leaf node evaluations will increase render time and reduce RMSE towards VRL ALC.

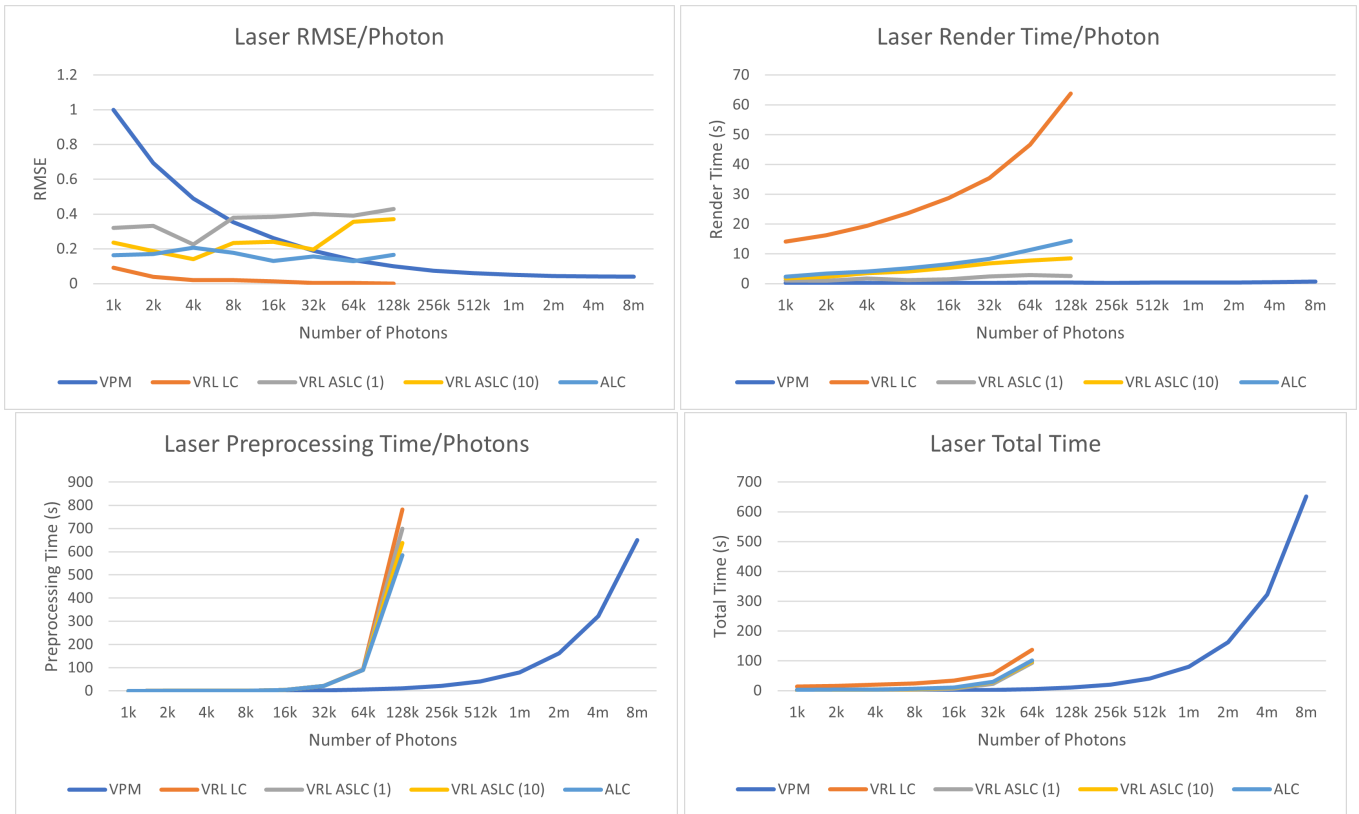


Figure 14: Scalability comparison between VPM, VRL LC, and VRL SLC at 1 and 10 leaf node evaluations for laser scene with non-linear camera rays (Figure 13).

SLC Convergence Speed The reduced render time of SLC over LC is appealing, but it requires many *spp* to converge into a noise-free image. Here we briefly show how quickly the RMSE progresses with more *spp* in the laser scene with non-linear camera rays. The SLC uses the atomic query method with uniform sampling which affects the error. Render time grows linearly with the number of *spp*. The 128 *spp* render took 3.3 minutes for SLC 1 and 11 minutes for SLC 10. The error of SLC 1 at 128 *spp* is equivalent to SLC 10 at 8 *spp* which took 43 seconds. A low maximum number of leaf node evaluations may be fast in terms of render time, but it results in greater error and visible singularities when using an

insufficient number of pixel samples. Particularly, notice in Figure 13 the SALC 10 render which shows more noise inside rectangles around the laser.

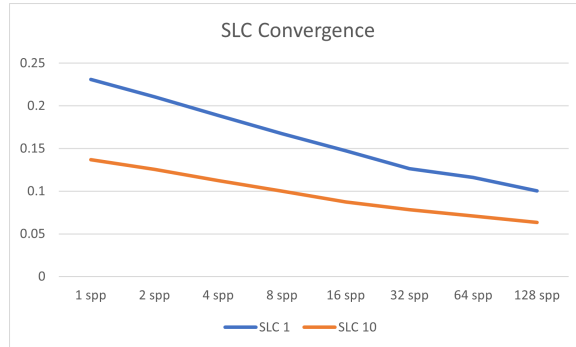


Figure 15: SLC convergence over increasing samples per pixel.

9.3 Mirage Scene

Figure 10 shows a superior mirage caused by the non-linear medium which causes light to bend towards the ground. An inverted reflection of the horizon can be observed. The image was rendered at a resolution of 1024 by 512 pixels and 16 spp.

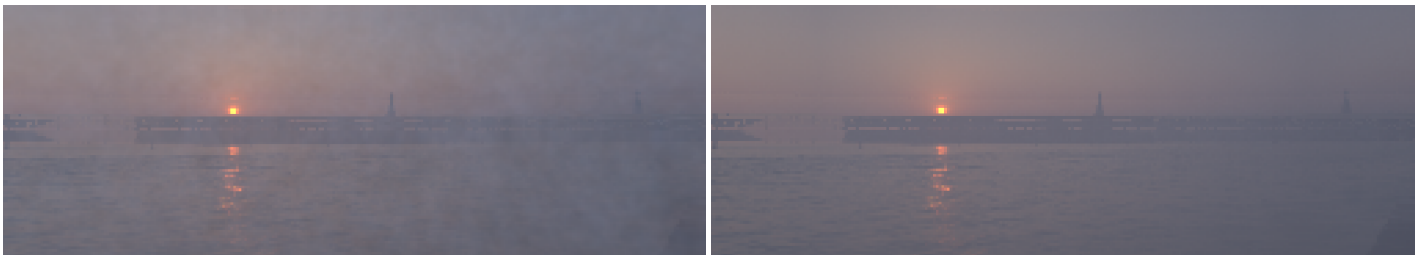


Figure 16: Superior mirage with thin fog layer using VPM and VRL LC.

Figure 16 shows the same superior mirage but with a layer of fog. Similar to the laser scene only medium-to-medium transport is rendered. The VPM reference render uses 8 million photons while the VRL LC render contains 128k virtual lights. Notice how, even with a large number of photons, there are still visible singularities for the VPM render while the VRL LC render shows a much smoother homogeneous scattering of light. This results in an RMSE lower than the real error once the VPM is finally converged. However, this may take multiple days to compute and we can already show significant improvements in total computation time using a suboptimal reference render for the VPM.

The overall shape of RMSE, render, preprocessing, and total time (see Figure 17) follows a similar pattern to the laser scene (Figure 14). Again, we observe that total time is dominated by the lightcut construction time for all VRL LC variations. Overall, render time for VRL ALC is less than VRL LC by a factor of 5-6. This improvement factor is nearly equivalent to what we observed from the laser scene.

When observing equal time renders between VPM and VRL LC we see that 8k VRLs result in an equivalent error to 4 million photons. The VRL render cost a total of 1.5 minutes while the VPM render cost 9 minutes. This shows a reduction

in total computation time by a factor of 6. Comparing VPM to VRL ALC we notice that the same 8k VRLs result in a similar error to 256k photons for VPM. The VRL ALC render took 16 seconds while the VPM render took 35 seconds, a performance gain with a factor of 2. Introducing even more noise using SLC the error is closer to 64k photons. SLC1 cost 2.5 seconds to total computation time while 64k photons took 9 seconds, an improvement factor of 3.

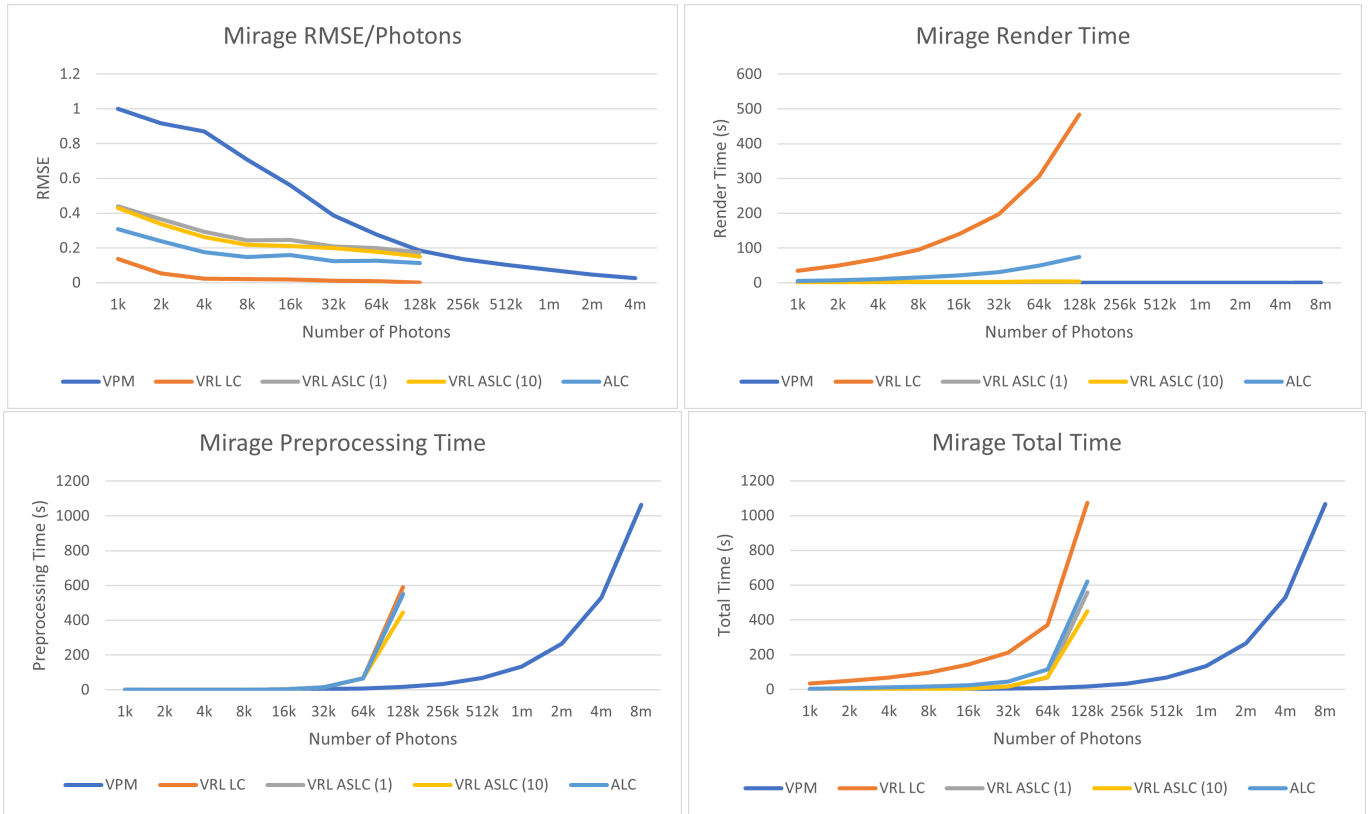


Figure 17: Scalability comparison between VPM, VRL LC, and VRL SLC at 1 and 10 leaf node evaluations for mirage scene with non-linear camera rays (Figure 16).

10 Discussion and Limitations

It was unfortunate to discover that the non-linear photon mapping method is biased and results in slightly darker images. This prevents us from measuring the degree of bias that VRLs may introduce in a non-linear media. However, we have shown side-by-side comparisons between reference VPM and VRL renders which shows that both methods converge to a highly similar result. Multiple scattering causes a spreading of light and generally does not include much detail. Only in the laser scene, we can observe the primary path that photons take throughout the media from indirect illumination alone. Even though the VRLs are straight line segments they do not cause visible artifacts on the curved path. From a theoretical standpoint, any VPL-based framework is a biased solution in non-linear media. However, the non-observable bias can be acceptable to achieve a more efficient total render time, particularly in media that is costly to traverse. From the experiments, we can conclude that VRLs can definitely be used in non-linear media and handle indirect multiply scattered illumination much more efficiently.

The primary benefit of VPM is its efficient rendering time which stays under a few seconds even with millions of photons. Across all scenes, this is significantly less than the standard VRL LC method which has to query each piecewise linear ray segment. Our adaptation of VRL LC into VRL ALC by querying the entire non-linear ray significantly reduces render time for VRLs by reducing the number of queries. The implementation is limited to uniform sampling which results in a higher RMSE and visible noise. However, it reduces render time such that the difference between using linear and non-linear camera rays becomes insignificant. This shows that with a slight adaption VRLs can be used efficiently in a non-linear media.

The primary downside of VPM is the need to sample millions to billions of photons to achieve a singularity-free result. In all scenes, we observe that enough VRLs can be gathered within a few seconds while VPM requires many hours. The preprocessing time is still significantly higher due to our inefficient lightcut construction method. The construction time in our case grows quadratically with the number of VRLs. Much more efficient methods could be used to significantly speed up this process, such as fast agglomerative clustering as proposed by Walter et al. [31].

VRLs alone cannot render the complete VRE and thus we combined this method with VPM for direct and caustic illumination in volumes and on surfaces. The Cornell box scene shows that the performance benefit over VPM in preprocessing time is non-existent as the surface photons take longer to trace than the required number of volume photons or VRLs. One way to solve this is to compute direct illumination using standard techniques, but this does not produce accurate results. However, this could be an acceptable alternative for previews or interactive applications.

Stochastic lightcuts are a great addition to lightcuts for VRLs. VRLs are arguably too good at computing multiple scattering within a single sample per pixel. In both the laser and mirage scene we can observe that even 1-4k VRLs result in a highly similar image to the final reference. This is reflected in a much higher rendering time compared to VPM. However, one can think of many scenarios where multiple samples per pixel are necessary such anti-aliasing or scenes with dielectric BSDFs. Thus, using SLC and the configurable number of maximum leaf node evaluations we can effectively spread out the computation of medium-to-medium transport over multiple pixel samples. Our implementation only consists of a basic importance sampling scheme entirely based on the intensity and length of the VRLs. One could think of more effective importance sampling schemes that, for example, take into account the geometric term G as proposed by Walter et al. [32].

11 Conclusion

While we cannot provably show the degree of bias that VRLs introduce in a non-linear environment we have visibly shown the difference. The results show that VRLs can be used in a non-linear environment and converge to the reference image using far fewer photons. This is especially advantageous in non-linear media where tracing photons from light sources can be expensive. With increased sample density, VRL converges quicker to the reference than VPM using many fewer emissions and memory footprint. Adapting the VRL to the camera ray contribution integration step to sample a single point along a curved ray significantly reduces the performance impact non-linear media has on the original VRL LC implementation. Additionally, we show that render time can be further reduced using stochastic lightcuts for VRLs. The advantage of making lightcuts for VRLs stochastic is that the computation of medium-to-medium transport can be effectively spread out over multiple pixel samples. Finally, we have shown that stochastic lightcuts for VRLs allow near interactive render time and close the gap between VPM and VRL render times for a significantly reduced error.

12 Future Work

The primary limitation of this research is the fact that the correctness of using VRLs in a non-linear media cannot be effectively measured. Furthermore, the methods we use here are all based on the RTE. Ament et al. [2] extended the RTE towards the RRTE which ensures the conservation of energy within a non-linear media. It also includes transient rendering and a wavelength-dependent index of refraction. To properly evaluate the correctness of VRLs in a non-linear environment, the VRL method should be extended toward the RRTE. This was deemed out of scope for a master thesis and thus it is left for future work.

Integrating the contribution of a VRL onto a curved ray increases noise as we are limited to uniform sampling. Uniform sampling is suboptimal as it may require hundreds of pixel samples to converge in a way that is not directly in our control. Importance sampling VRL contribution requires finding the closest point between the VRL and the camera ray. This is trivial between two straight line segments but becomes more complex between a straight line segment and a curved piecewise-linear segment. One could naively go over all piecewise-linear segments to find the closest point, but this would cause the time to estimate the VRL contribution to grow linearly with the resolution of the non-linear method. We leave importance sampling along curved line segments for future work.

It has been shown that VRLs can render multiply scattered light within linear and non-linear participating media much more efficiently. However, in scenes where the volume photon map contains mostly direct rather than indirect illumination the performance benefit diminishes as VRLs cannot handle direct illumination. In the worst-case preprocessing time degrades to the same preprocessing time for VPM as shown using the Cornell box scene. It would be interesting to explore more efficient methods to sample direct illumination on surfaces through non-linear media.

References

- [1] Mitsuba 2 - a retargetable forward and inverse renderer.
- [2] Marco Ament, Christoph Bergmann, and Daniel Weiskopf. Refractive radiative transfer equation. *ACM Transactions on Graphics (TOG)*, 33(2):1–22, 2014.
- [3] Marc Berger, Terry Trout, and Nancy Levit. Ray tracing mirages. *IEEE Computer Graphics and Applications*, 10(3):36–41, 1990.
- [4] Benedikt Bitterli, Chris Wyman, Matt Pharr, Peter Shirley, Aaron Lefohn, and Wojciech Jarosz. Spatiotemporal reservoir resampling for real-time ray tracing with dynamic direct lighting. *ACM Transactions on Graphics (TOG)*, 39(4):148–1, 2020.
- [5] Chen Cao, Zhong Ren, Baining Guo, and Kun Zhou. Interactive rendering of non-constant, refractive media using the ray equations of gradient-index optics. In *Computer Graphics Forum*, volume 29, pages 1375–1382. Wiley Online Library, 2010.
- [6] Subrahmanyan Chandrasekhar. *Radiative Transfer*. Courier Corporation, 1960.
- [7] Carsten Dachsbacher, Jaroslav Krivánek, Miloš Hašan, Adam Arbree, Bruce Walter, and Jan Novák. Scalable realistic rendering with many-light methods. In *Computer Graphics Forum*, volume 33, pages 88–104. Wiley Online Library, 2014.
- [8] Eduard Gröller. Nonlinear ray tracing: Visualizing strange worlds. *The Visual Computer*, 11(5):263–274, 1995.
- [9] Diego Gutierrez, Adolfo Munoz, Oscar Anson, and Francisco J Seron. Non-linear volume photon mapping. In *Rendering Techniques*, pages 291–300, 2005.
- [10] Diego Gutierrez, Francisco J Seron, Oscar Anson, and Adolfo Muñoz. Chasing the green flash: a global illumination solution for inhomogeneous media. In *Proceedings of the 20th spring conference on Computer graphics*, pages 97–105, 2004.
- [11] Diego Gutierrez, Francisco J Seron, Adolfo Munoz, and Oscar Anson. Simulation of atmospheric phenomena. *Computers & Graphics*, 30(6):994–1010, 2006.
- [12] Miloš Hašan, Jaroslav Krivánek, Bruce Walter, and Kavita Bala. Virtual spherical lights for many-light rendering of glossy scenes. In *ACM SIGGRAPH Asia 2009 papers*, pages 1–6. 2009.
- [13] Miloš Hašan, Fabio Pellacini, and Kavita Bala. Matrix row-column sampling for the many-light problem. In *ACM SIGGRAPH 2007 papers*, pages 26–es. 2007.
- [14] Ivo Ihrke, Gernot Ziegler, Art Tevs, Christian Theobalt, Marcus Magnor, and Hans-Peter Seidel. Eikonal rendering: Efficient light transport in refractive objects. *ACM Transactions on Graphics (TOG)*, 26(3):59–es, 2007.
- [15] Wojciech Jarosz, Derek Nowrouzezahrai, Robert Thomas, Peter-Pike J Sloan, and Matthias Zwicker. Progressive photon beams. *ACM Trans. Graph.*, 30(6):181, 2011.
- [16] Wojciech Jarosz, Matthias Zwicker, and Henrik Wann Jensen. The beam radiance estimate for volumetric photon mapping. In *ACM SIGGRAPH 2008 classes*, pages 1–112. 2008.

- [17] Henrik Wann Jensen and Per H Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 311–320, 1998.
- [18] James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 143–150, 1986.
- [19] Alexander Keller. Instant radiosity. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 49–56, 1997.
- [20] Daqi Lin, Chris Wyman, and Cem Yuksel. Fast volume rendering with spatiotemporal reservoir resampling. *ACM Trans. Graph.*, 40(6), dec 2021.
- [21] Daqi Lin and Cem Yuksel. Real-time stochastic lightcuts. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 3(1):1–18, 2020.
- [22] Qi Mo, Hengchin Yeh, and Dinesh Manocha. Tracing analytic ray curves for light and sound propagation in non-linear media. *IEEE transactions on visualization and computer graphics*, 22(11):2493–2506, 2015.
- [23] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. In *Computer Graphics Forum*, volume 37, pages 551–576. Wiley Online Library, 2018.
- [24] Jan Novák, Derek Nowrouzezahrai, Carsten Dachsbacher, and Wojciech Jarosz. Progressive virtual beam lights. In *Computer Graphics Forum*, volume 31, pages 1407–1413. Wiley Online Library, 2012.
- [25] Jan Novák, Derek Nowrouzezahrai, Carsten Dachsbacher, and Wojciech Jarosz. Virtual ray lights for rendering scenes with participating media. *ACM Transactions on Graphics (TOG)*, 31(4):1–11, 2012.
- [26] Jan Novák, Andrew Selle, and Wojciech Jarosz. Residual ratio tracking for estimating attenuation in participating media. *ACM Trans. Graph.*, 33(6):179–1, 2014.
- [27] Simon Premože, Michael Ashikhmin, Jerry Tessendorf, Ravi Ramamoorthi, and Shree Nayar. Practical rendering of multiple scattering effects in participating media. In *Proc. of Eurographics Symposium on Rendering*, volume 2, pages 363–374. Citeseer, 2004.
- [28] Xin Sun, Kun Zhou, Eric Stollnitz, Jiaoying Shi, and Baining Guo. Interactive relighting of dynamic refractive objects. In *ACM SIGGRAPH 2008 papers*, pages 1–9. 2008.
- [29] Eric Veach. Non-symmetric scattering in light transport algorithms. In *Eurographics Workshop on Rendering Techniques*, pages 81–90. Springer, 1996.
- [30] Nicolas Vibert, Adrien Gruson, Heine Stokholm, Troels Mortensen, Wojciech Jarosz, Toshiya Hachisuka, and Derek Nowrouzezahrai. Scalable virtual ray lights rendering for participating media. In *Computer Graphics Forum*, volume 38, pages 57–65. Wiley Online Library, 2019.
- [31] Bruce Walter, Kavita Bala, Milind Kulkarni, and Keshav Pingali. Fast agglomerative clustering for rendering. In *2008 IEEE Symposium on Interactive Ray Tracing*, pages 81–86. IEEE, 2008.
- [32] Bruce Walter, Sebastian Fernandez, Adam Arbree, Kavita Bala, Michael Donikian, and Donald P Greenberg. Lightcuts: a scalable approach to illumination. In *ACM SIGGRAPH 2005 Papers*, pages 1098–1107. 2005.

- [33] Can Yuksel and Cem Yuksel. Lighting grid hierarchy for self-illuminating explosions. *ACM Trans. Graph.*, 36(4):110–1, 2017.
- [34] Cem Yuksel. Stochastic lightcuts. 2019.
- [35] Ye Zhao, Yiping Han, Zhe Fan, Feng Qiu, Yu-Chuan Kuo, Arie E Kaufman, and Klaus Mueller. Visual simulation of heat shimmering and mirage. *IEEE transactions on visualization and computer graphics*, 13(1):179–189, 2006.

13 Appendix

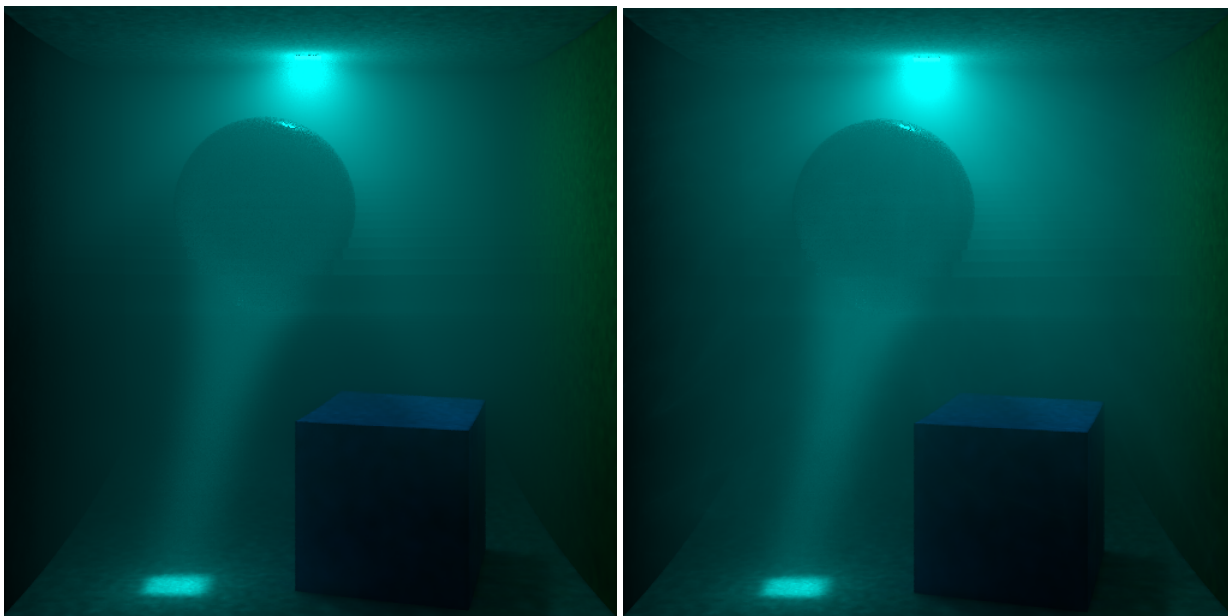


Figure 18: HD render of the nonlinear Cornell Box scene using VPM (top) and VRL (bottom).



Figure 19: Mirage with fog scene using VRL with a resolution of 1024 by 512 and 1 spp. Only 32k VRLs were necessary.

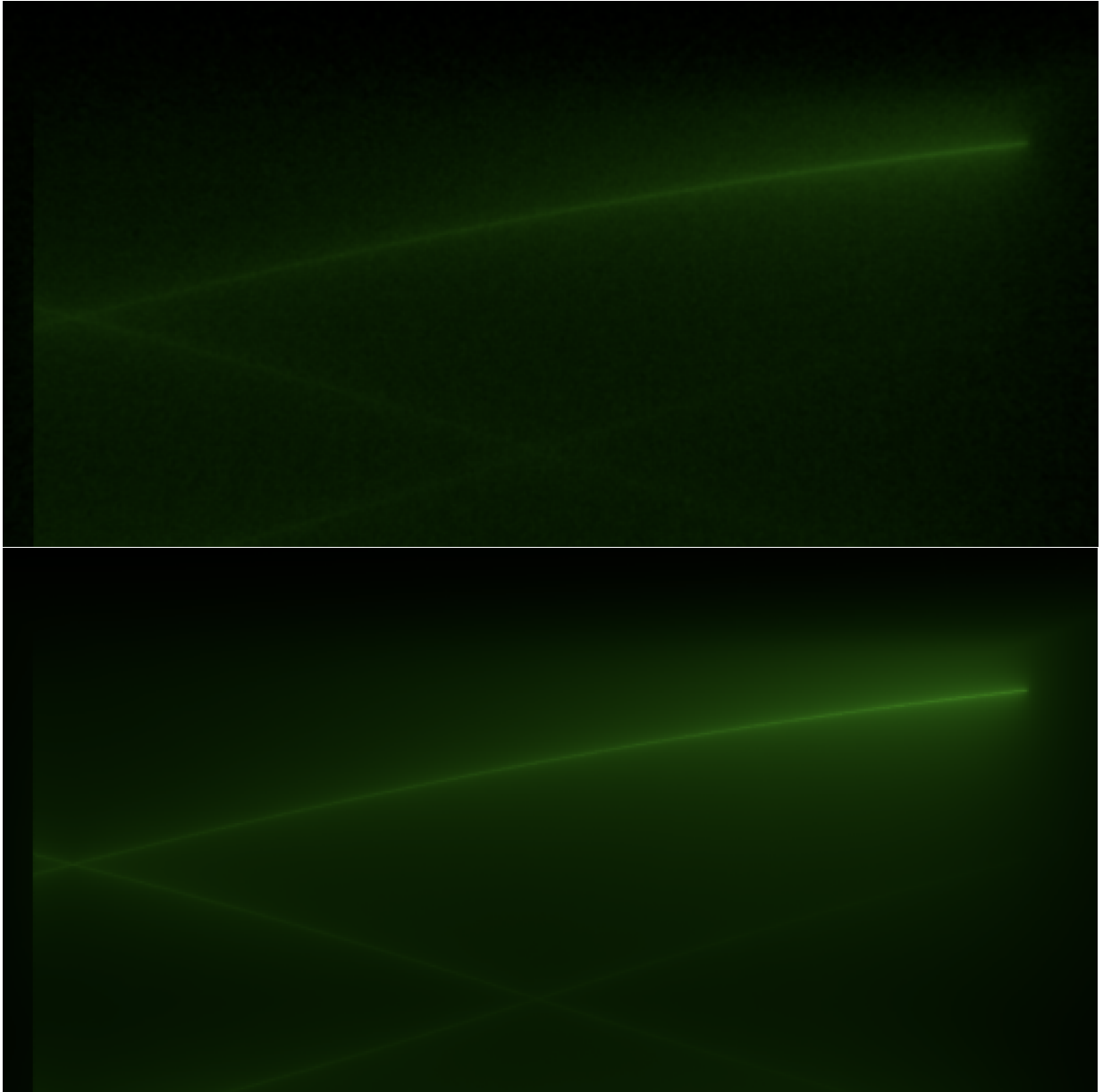


Figure 20: Laser scene with linear camera rays using VPM (top) and VRL (bottom).

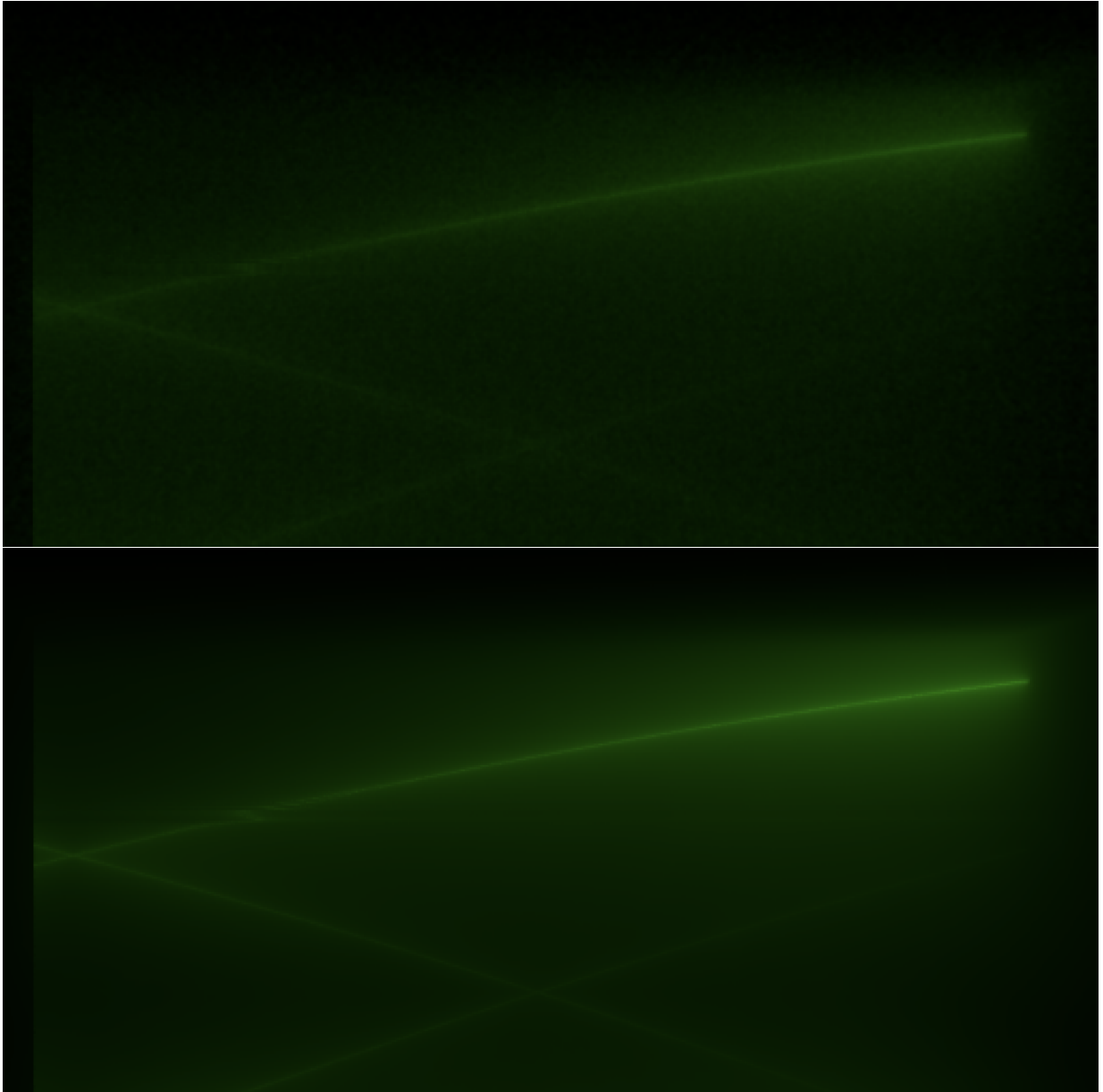


Figure 21: Laser scene with non-linear camera rays using VPM (top) and VRL (bottom).