**Lighting Caustics**

Final Year Dissertation

Supervisor: Benjamin Kenwright

Student: Zilun Zhang

undergraduate

Computer Science

**DECLARATION**

I, *your name* confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of other authors in any form (e.g., ideas, equations, figures, text, tables, programs) are properly acknowledged at any point of their use. A list of the references employed is included.

Signed: *Zilun Zhang*

Date: *date of submission*

**Abstract**

**Table**

1.Introduction

1.1 Project Description

The phenomenon of caustics is the separation of colours that occurs when light passes through or touches an unsmooth object due to the different speeds at which different wavelengths (colours) of light travel through the medium, resulting in different angles of refraction of the light. Thus, the phenomenon of focal dispersion can be seen everywhere in everyday life, from the shimmering sensation of sunlight shining into the surface of water to the light spots produced by sunlight shining into a glass of water.

The phenomenon of focal dispersion has a wide range of uses in other areas.

In architectural design, the understanding and application of focal dispersion can help architects to optimize the interior light environment. For example, when designing structures such as skylights and glass curtain walls, the focal dispersion phenomenon can be considered to achieve effective guidance and distribution of light, improving indoor comfort and energy efficiency. [2]

Some organisms, such as squid and fish, can regulate the refraction and reflection of light by controlling certain structures in their bodies, thereby creating the phenomenon of focal dispersion. This phenomenon has some research value in the field of biology and can help us to better understand the light perception and adaptive capabilities of these organisms. [3]

The implementation of caustics is not easy and can be overlooked because in some cases the phenomenon is not noticeable, for example in some simple imaging systems where the focal dispersion effect is small and therefore negligible because the light travels over short distances. And in some applications, solving the focal dispersion problem may require additional optical components and design, which may add cost and system complexity. Yet systems are more tolerant of caustics, so optimizing for it may not be a priority. For example, in some simple observation or monitoring systems, the requirements for image quality may be low and therefore a degree of focal dispersion may be acceptable.

In recent years, the importance of focal dispersion phenomena will receive more and more attention as scientific and technological developments lead to a better understanding of optical phenomena.

However, as technology advances, so does the technology of ray tracing, so that using this technology provides us with an accurate simulation of light propagation in the real world.

1.2 Aim

Objective 1: Background research

Background research on the research topic project to better understand the chosen topic and the scope of the project. It can help to clarify next goals.

Objective 2: Survey

Survey of existing algorithms and methods: algorithms that currently exist and analysis of their strengths and weaknesses. From there, This can help in choosing the right method best suited.

Objective 3: Simple work

Construct simple non-real time ray-tracing simulations and simulate the effects of reflection and refraction using the underlying algorithms.

Objective 4: Add caustics

In this objective, the underlying focal dispersion algorithm for objects in ray tracing is investigated and it is expected that caustics rays can be added.

Objective 5: More work

Add more models for Objective 4 (different materials, different surfaces, etc.). Thus the variation in speed of ray tracing is studied.

Objective 6: Optimized for ray tracing

In the process, more optimisations for ray tracing are investigated, leading to faster rendering in Objective 5.

Objective 7：Evaluate

Evaluate algorithms for 6 and perform incremental development. Adding movable scenes, movable cameras, and real-time ray tracing rendering.

1.3 Limit

Due to the high reliance on computer hardware for this project, hardware limitations were one of the issues that had to be considered. Better/newer GDP technologies could make rendering faster, and for the latest NVIDIA RTX 40, DLSS ray tracing could be used, a technology that increases the speed of ray tracing rendering.

In my work on Goal 1, I found that there are various implementation algorithms for ray tracing and focal dispersion, including those of the current mainstream rendering engines, which led to an excessive amount of time spent analysing the methods.

For Objectives 5 and 6 in this paper, the scope of this project could be increased or decreased in this session, including optimising the novelty of the solution and optimising the real-time ray tracing, due to the excessive hardware limitations and time involved with the computer.

1.4 Document Structure

A

2. Background

This article BACKGROUND will cover ray tracing, types of real time ray tracing and the concept of focal dispersion, as well as relevant algorithms and methods for implementation, and finally outline methods for optimizing ray tracing.

2.1 Raytracing

Ray tracing is a computer rendering technique that simulates the propagation and interaction of light in a 3D scene to produce realistic images. In ray tracing, light is emitted from a viewpoint into the scene, following a path through the scene and intersecting with objects. By calculating properties such as colour, shading, reflection and refraction at the point where the light intersects the object, and using these as the colour value of the pixel. In this way, high quality, realistic images can be generated.

In summary, the realistic propagation behavior of light is simulated to track the light for rendering. 【4】【1】

In physics terms, ray tracing is a method of calculating the path of particles through different objects (with different propagation speeds, absorption properties and reflective surface areas). 【5】

And there are different ways to differentiate and implement ray tracing

2.11 Classic Ray Tracing

This ray tracing is very simple and it is suitable for generating static images. The basic ray tracing algorithm simulates the propagation of light through the scene by emitting light from pixels. Once the applicable ray intersects an object in the scene, the algorithm calculates the colouring and shading of the object's surface, as well as the path of the light after reflection and refraction. The final pixel colours are obtained from these calculations.【6】

First, the ray is emitted from the camera, passes through a pixel point on the screen and enters the 3D scene, after which it is determined whether this ray has collided with an object in the scene.

If there is no collision, it returns to the background.

In reality, refraction and reflection occur when a ray of light hits an object, and in classical ray tracing, the direction of reflection is also calculated when the ray hits an object. Once the direction of reflection has been obtained, the light does not continue in a new direction and this light is used for the pixel colour value. If this ray intersects another object then the ray from the light source to the point of collision is blocked and the colour value of this pixel point in the shadow is then calculated according to the rendering equation; if no intersection occurs then the light source can be directed to the point of collision and the colour of the pixel point in the ray is then calculated.

This process is then repeated and the image is finally rendered.

However, this method lacks the phenomenon of refraction and can only accept light reflected from the light source.【6】【7】

2.12 Whitted-style Ray Tracing（Recursive Ray Tracing）

Recursive ray tracing is also a branch that evolved from classical ray tracing, it is basically the same principle as classical ray tracing, but instead of abandoning the extended ray after finding a point of contact, this point of collision is used as the new starting point of the ray, then the direction of reflection of the ray is calculated at this point as the new direction of the ray, which is then implemented with a recursive function that continues to trace this ray, i.e. continues to find this new ray's nearest point of collision. These steps are repeated until the ray meets the light source or wall.

The disadvantage of this ray tracing is that if the light source is too small, then a lot of the light will be concentrated on the wall and the final rendered image will have a lot of noise. [6]

It is also classified as forward ray tracing and backward ray tracing.

Backward ray tracing is a more common method of ray tracing. In this method, light is emitted from the camera, passes through objects in the scene, and eventually reaches the light source. However, this method may waste some computational resources, as many rays may not end up reaching the light source. [6] [7] [8]

Forward ray tracing is a method of emitting light from a light source. In this method, light travels from the light source, along objects in the scene, and eventually reaches the camera. However, forward ray tracing is usually less efficient as a large amount of light may not reach the camera, resulting in wasted computational resources. [6][7][8]

different Classification of ray tracing

2.13 Global Illumination

Global Illumination is a model of lighting in computer graphics that simulates multiple scattering of light in a scene, including both direct and indirect illumination. In global illumination it is necessary to calculate the sum of all the light received by each pixel point in the whole scene, which usually requires processing many light-object intersection calculations and is therefore computationally expensive. Furthermore, when simulating indirect lighting and complex materials, it is possible to produce very realistic images, at the same time increasing rendering time and computational complexity. [8][7]

2.14 real-time ray tracing

Real-time ray tracing aims to enable interactive or real-time ray tracing rendering through optimized algorithms and hardware acceleration. Real-time ray tracing typically sacrifices some rendering quality in exchange for higher frame rates and interactivity. In recent years, real-time ray tracing has become increasingly common in games and real-time applications as the performance of graphics hardware has increased. For example, NVIDIA's RTX graphics card series accelerates the computational process of real-time ray tracing by introducing dedicated ray tracing hardware (RT cores). [9] [10]

The term "real-time" in this context means that the image is rendered fast enough so that the user can observe the movement, deformation, or lighting changes of objects in a real-time scene.

2.15 summery

With the development of technology, the public is more in pursuit of more realistic and interesting videos, images, however ray tracing can produce high quality, realistic images and animations due to the accurate calculation of the light propagation path, which can produce accurate lighting effects such as shadow details and reflection accuracy, and with the continuous development of computer hardware and algorithms, the efficiency of ray tracing algorithms continues to increase. It can be used for real-time rendering and interactive applications, so it is widely used in computer graphics, computer games, film and animation production, making it more sought after by the public.

2.2 Caustics

Caustics are optical phenomena that occur when light passes through or touches a transparent or translucent object with curves and refractive or reflective properties (e.g. water, glass, etc.) and forms a concentration of brightness on the receiving surface. On the surface of water, sunlight is reflected and refracted, creating rippling bright patterns. An example is the light spots and ripples at the bottom of a swimming pool.

The most confusing phenomenon is dispersion, which is caused by the refractive index of an object changing with wavelength, for example a prism breaking up white light into coloured light.【11】

2.2.1 Formation processes (mathematics)

The formation process of focal Caustics involves several physical phenomena, such as refraction, reflection, and dispersion.

1. Refraction（Snell's Law）

When light rays pass from one medium into another, the law of refraction describes the relationship between the incident and refracted rays.

Mathematical equations are as follows：

2. Reflection Law

The law of reflection describes the relationship between the incident and reflected rays of light as they pass from one medium into another. The direction of the incident and reflected rays is governed by the law of reflection. [12]

Mathematical equations are as follows：

3. Fresnel equations

Describes the behavior of reflection and refraction of light rays when they are directed from an interface between media into another medium. The direction of the incident and refracted rays is governed by Snell's law.[12]

Parallel light equation:

T∥​=24n1​n2​cosθi​cosθt​​/(n1​cosθi​+n2​cosθt​)

4. Radiative Transfer Equation

In computer graphics, the equations for light transmission describe the processes of propagation, absorption, scattering and emission of light in a scene.

Mathematical equations are as follows：

Reference

[1] https://en.wikipedia.org/wiki/Ray\_tracing\_(graphics)

[2] Mardaljevic, J. (1999). Daylight simulation: validation, sky models and daylight coefficients (Doctoral dissertation, De Montfort University).

[3] Johnsen, S. (2001). Hidden in plain sight: the ecology and physiology of organismal transparency. *The Biological Bulletin*, *201*(3), 301-318.

[4] Akenine-Mo, T., Haines, E., & Hoffman, N. (2018). Real-time rendering

【5】<https://en.wikipedia.org/wiki/Ray_tracing_(physics)>

【6】Whitted, T. (2005). An improved illumination model for shaded display.

【7】Peddie J. (2019) Ray Tracing: A Tool for All.

【8】Akenine-Mo, T., Haines, E., & Hoffman, N. (2018). Real-time rendering.

【9】<https://developer.nvidia.com/rtx/ray-tracing>

【10】Haines, E., & Akenine-Möller, T. (Eds.). (2019). *Ray Tracing Gems: High-Quality and Real-Time Rendering with DXR and Other APIs*. Apress.

【11】<https://en.wikipedia.org/wiki/Caustic_(optics)>

【12】https://en.wikipedia.org/wiki/Fresnel\_equations

Yang, X., & Ouyang, Y. (2021). Real-Time Ray Traced Caustics. *Ray Tracing Gems II: Next Generation Real-Time Rendering with DXR, Vulkan, and OptiX*,