

# Feasibility Study of Real Time Path Tracing

*Or: How Much Noise Is Too Much?*

**Sven-Hendrik Haase**

A thesis presented for the degree of  
*Bachelor of Computer Science*



Department of Informatics  
University of Hamburg  
Germany  
2015-20-08

Primary Supervisor: Prof. Dr. rer. nat. Leonie DRESCHLER-FISCHER  
Secondary Supervisor: Prof. Dr. Thomas LUDWIG

# Abstract

This study aims to investigate the viability of a physically-based technique called **path tracing** in lieu of or in corporation with classical techniques in interactive media such as video games and visual effects tools.

Real time path tracing has been prohibitively expensive in regards to computational complexity. However, modern GPUs and even CPUs have finally gotten fast enough for real time path tracing to become a viable alternative to traditional real time approaches to rendering. Based on that assumption, this thesis presents the idea, algorithm and complexity behind path tracing in the first part and extrapolates feasibility and suitability of real time path tracing on consumer hardware according to the current state of technology and trends in the second part.

As part of the research, the author has implemented a path tracing 3D engine in modern C++ in order to empirically test the assumptions made in this thesis. The study found path tracing to be a viable rendering technique for average commodity hardware in approximately 4 years.

# Acknowledgments

I would like to express my sincere gratitude to the teachers throughout school and university for the knowledge they've passed on.

I thank my friends for the laughs, horrible mistakes and awesome successes we shared with one another.

Furthermore, none of this would have been possible without the incredible efforts and love of my parents who have supported me throughout the years and enabled me to live a carefree life until I was ready to fend for myself.

Lastly, but certainly not least, I would like to declare my gratefulness to Alisa, whose endless love has given my life a new meaning.

*We all make choices in life, but in the end, our choices make us.*

*Andrew Ryan*

# Contents

<b>1</b>	<b>Introduction</b>	<b>7</b>
1.1	Motivation . . . . .	7
<b>2</b>	<b>Real Time Path Tracing Explained</b>	<b>9</b>
2.1	Theoretical Basis . . . . .	9
2.2	Path Tracing in Comparison to Other Techniques . . . . .	9
2.2.1	Comparison to Rasterization . . . . .	9
2.2.2	Comparison to Other Algorithms . . . . .	9
2.3	History of Path Tracing . . . . .	9
2.4	Current State of Technology . . . . .	11
<b>3</b>	<b>Research</b>	<b>13</b>
3.1	Results . . . . .	13
3.2	Evaluation . . . . .	13
<b>4</b>	<b>Conclusion</b>	<b>15</b>
4.1	Outlook . . . . .	15

# Acronyms

**AABB** axis-aligned bounding box

**AO** ambient occlusion

**BRDF** bidirectional reflection distribution function

**BSP** binary space partitioning

**CPU** central processing unit

**CSG** constructive solid geometry

**FPS** frames per second

**GI** global illumination

**GPU** graphics processing unit

**HDR** high dynamic range

**MLT** Metropolis light transport

**PBR** physically based rendering

**RT** real time

**RGB** red-green-blue

**SIMD** single instruction multiple data

**SPP** samples per pixel

**SSAO** screen-space ambient occlusion

# 1 Introduction

As part of the quest for ever-improving game graphics, researchers, graphics hardware developers and video game developers alike have been coming up with more and more convoluted and technically challenging ways of improving the graphics in interactive media such as games and visualizations in order to give users a deeper sense of immersion or to provide special effects artists with faster feedback.

While rendering techniques are currently shifting from the traditional fixed pipeline approach towards the new, fully programmable approach that lets developers implement deferred renderers that can more closely mimic reality by using multiple combined shading and lighting algorithms and rendering the scene multiple times for different buffers, the fundamental concept of rasterization-based rendering has largely remained the same.

The real world photon-collecting approach that actual cameras use has so far not been adopted for interactive media by the industry in any capacity because the computational cost has historically been prohibitively expensive. It is, however, used extensively (and has been in use since decades) for offline, non-interactive rendering of computer-generated movies and visualizations of scientific simulations.

This study assumes that the next logical step for the industry will be to adopt this method for real time media as well. For the purpose of this thesis, a renderer is considered *real time* when it manages to render a frame within  $16.67ms$  since that equals 60 frames per second (FPS) which is the current de-facto standard refresh rate for most available computer screens. Conversely, a renderer is called *offline* when it is not designed for interactive rendering which usually means that it will render an image or a batch of images over the course of a few days. The differences of real time and offline path tracing renderers will be explained in the next chapter.

## 1.1 Motivation

Real time path tracing (and physically based rendering in general) offers many benefits over traditional real time rendering methods such as better visuals and simpler implementation but also allows for completely new types of graphics such as realistic caustics [1] and even light dispersion [2] (using a prism, for instance) since path tracers might simulate wavelengths instead of plain red-green-blue (RGB) colors. Modern video

games tend to rely on a growing number of tricks to keep them visually appealing as the consumer grows more demanding. They're called *tricks* in this study because they merely trick the beholder into seeing something that appears to be physically accurate when it is, in fact, not the result of a physically-based calculation and as such this study aims to keep tricks and emergent phenomena separated by language. Some notable tricks include screen-space ambient occlusion (SSAO) [3], motion blur [4], lens flares [5], chromatic aberration [6], depth of field [7] and light mapping [8].



## 2 Real Time Path Tracing Explained

This chapter will explain the concepts, mathematics and algorithms behind path tracing, how real time path tracing differs from offline path tracing and what the difference in speed means for quality. It will also explain how path tracing differs from rasterization and other global illumination techniques.

### 2.1 Theoretical Basis

This explanation will Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 2.2 Path Tracing in Comparison to Other Techniques

#### 2.2.1 Comparison to Rasterization

#### 2.2.2 Comparison to Other Algorithms

### 2.3 History of Path Tracing

As with so many things in computer science and science in general, the modern idea of physically based rendering using path tracing builds upon many important past discoveries and algorithms such as ray tracing and ray casting. Arthur Appel is generally

credited as being the father of *ray casting* as he was the first to describe the algorithm in a 1968 paper [9].

Ray casting is an important idea needed for *ray tracing* which was first published in a paper in 1980 by Turner Whitted [10].



Figure 2.1: Turner Whitted’s original 1980 [10] image showing off the usage of ray tracing for reflection, refraction and shadows.

Building upon ray tracing, an improved algorithm was published in 1986 by James T. Kajiya which used ray tracing combined with a Monte Carlo algorithm in order to create a new algorithm that was called *Monte Carlo ray tracing* [11]. Nowadays, Monte Carlo ray tracing is better known as *path tracing*.

It took another decade for path tracing to become the physically based rendering approach that it is known for today. In 1996, Eric Lafortune improved the algorithm by suggesting the usage of bidirectional path tracing [12] and finally the Metropolis light transport was suggested in 1997 by Eric Veach and Leonidas J. Guibas [13] to improve performance in complex scenes.

This was the last notable improvement to the algorithm, though many micro optimizations have since been published. All of these achievements and improvements are generally collapsed into the term *path tracing* since they do not diverge from the general algorithm but instead improve upon it.

It took longer still for the industry to become interested in path tracing. The early interest in ray tracing was of mostly academical and recreational nature. One of the most notable creations of the early days of ray tracing is *The Juggler* created and published by Eric Graham in 1986 [14] on an Amiga 1000. It was a pre-rendered animation using

ray tracing. Eric Graham stated that it took the Amiga 1 hour to render each frame [14].



Figure 2.2: Eric Graham's Juggler

While the animation seems very primitive compared to the animations of today, it was exceptional at the time. Ernie Wright's statement about *The Juggler* provides some contemporary context:

Turner Whitted's paper (1980) is widely regarded as the first modern description of ray tracing methods in computer graphics. This paper's famous image of balls floating above a checkerboard floor took 74 minutes to render on a DEC VAX 11/780 mainframe, a \$400,000 computer. *The Juggler* would appear a mere six years later, created and displayed on a \$2000 Amiga. ([14])

The first feature-length computer-animated film, *Toy Story*, released in 1995 [15], is sometimes miscredited as being the first film using a ray tracing-like algorithm. However, it actually used traditional scanline rendering. The first feature-length film using ray tracing, *Cars*, was released much later, in 2006 [16] [17] and started a wave of interest in the movie industry.

## 2.4 Current State of Technology

lol Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## 3 Research

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 3.1 Results

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 3.2 Evaluation

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis

egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## 4 Conclusion

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 4.1 Outlook

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## List of Figures

2.1	Turner Whitted's original 1980 [10] image showing off the usage of ray tracing for reflection, refraction and shadows. . . . .	10
2.2	Eric Graham's Juggler . . . . .	11



## List of Tables

## Bibliography

- [1] Wikipedia. Caustic (optics) — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [2] Wikipedia. Dispersion (optics) — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [3] Wikipedia. Screen space ambient occlusion — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [4] Wikipedia. Motion blur — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [5] Wikipedia. Lens flare — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [6] Wikipedia. Chromatic aberration — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [7] Wikipedia. Depth of field — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [8] Wikipedia. Lightmap — wikipedia, the free encyclopedia, 2015. [Online; accessed 16-September-2015].
- [9] Arthur Appel. Some techniques for shading machine renderings of solids. In *Proceedings of the April 30–May 2, 1968, Spring Joint Computer Conference*, AFIPS '68 (Spring), pages 37–45, New York, NY, USA, 1968. ACM.
- [10] Turner Whitted. An improved illumination model for shaded display. *Commun. ACM*, 23(6):343–349, June 1980.
- [11] James T. Kajiya. The rendering equation. In *Computer Graphics*, pages 143–150, 1986.

- [12] Eric Lafortune. Mathematical models and monte carlo algorithms for physically based rendering. Technical report, 1996.
- [13] Eric Veach and Leonidas J. Guibas. Metropolis light transport. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 65–76, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.
- [14] Ernie Wright. The juggler, 1998. [Online; accessed 19-September-2015].
- [15] Wikipedia. Toy story — wikipedia, the free encyclopedia, 2015. [Online; accessed 19-September-2015].
- [16] Wikipedia. Cars (film) — wikipedia, the free encyclopedia, 2015. [Online; accessed 19-September-2015].
- [17] P.H. Christensen, J. Fong, D.M. Laur, and D. Batali. Ray tracing for the movie ‘cars’. In *Interactive Ray Tracing 2006, IEEE Symposium on*, pages 1–6, Sept 2006.