Abstract

Type:	Advanced Seminar Thesis
Author:	Wöllert, Thomas (Dipl.-Inf. (FH))
Titel:	Realtime and Interactive Ray Tracing, Basics and Latest Developments
Date:	22nd June 2006
Number of Pages:	66
Field of Study:	Master of Science - Computer Graphics and Image Processing
University:	Munich University of Applied Sciences, Germany
Advisor:	Prof. Dr. A. Nischwitz

Ray Tracing was first described by Arthur Appel in 1968. Due to its high rendering time it is almost solely used in pre-rendered motion pictures featuring realistic illumination, reflection and refraction. During the past years big strides have been made to prepare ray tracing for realtime applications like computer games and virtual dynamic environments.

This semester thesis starts with an explanation of the principles of how ray tracing works, presenting the terms and different rendering styles (i.e. recursive, distribution etc.). Afterwards reasons, why ray tracing is not already used in todays graphics cards, as well as pros and cons refering to common rasterization are discussed.

Section 3 focuses on different methods to speed up the ray tracing process. This can be accomplished by reducing the number of rays or by using acceleration structures (i.e. uniform grid, kd-tree, etc.) to improve the intersection tests, taking up most of the time in the rendering process. Additional information on the latest developments regarding acceleration structures especially designed for dynamic scenes, can be found in Chapter 4.

With the described tools it is possible to realize different approaches aiming at realtime ray tracing, which are presented in Chapter 5 (CPU, hybrid CPU-GPU, GPU, special purpose hardware).

A special focus is laid upon GPU ray tracers in Section 6, explaining different approaches and their problems. A benchmark visualizing the results concludes this chapter.

Vital for the mass marketing of ray tracing is an easy-to-use programming interface. One approach with that potential is described in Part 7 of this document: OpenRT Realtime Ray Tracing API, featuring an OpenGL-like programming language.

Specific applications of ray tracing are described in Section 8, presenting graphical (i.e. massively complex models) and non-graphical (i.e. collision detection, artificial intelligence) examples. The final Part 9 concludes this document.

Keywords: ray tracing, interactive rendering, programmable graphics hardware, GPU, acceleration structures