

# Echtzeit Ray Tracing und Interaktive Globale Beleuchtungssimulation\*

Realtime Ray Tracing and Interactive Global Illumination

Ingo Wald, University of Utah, SaarLB Wissenschaftspreis 2004

**Zusammenfassung** Interaktive 3D-Graphik basiert heutzutage fast ausschließlich auf dem Rasterisierungsverfahren, welches aber in Hinblick auf Bildqualität zunehmend an seine Grenzen stößt. Eine Alternative zur Rasterisierung ist das Ray Tracing, welches zwar allgemein für bessere Bildqualität bekannt ist, aufgrund hoher Rechenanforderungen aber bisher im Allgemeinen als unvereinbar mit interaktiver Performanz galt. Im Rahmen der Dissertation werden mehrere neue Verfahren vorgestellt, die es erstmalig ermöglichen, auch auf Standard-PCs und PC Clustern Echtzeit Ray Tracing zu realisieren, und dies insbesondere auch für die interaktive globale Beleuchtungssimulation zu verwenden.

**Summary** Interactive 3D Graphics today is almost entirely dominated by triangle rasterization, which however becomes increasingly problematic with respect to therewith achievable image quality. An alternative to triangle rasterization is ray tracing, which is generally known for its superior image quality, but due to its high computational demands is usually perceived to be incompatible with interactivity. This thesis presents several new techniques that finally allow for achieving realtime ray tracing performance on PCs and PC clusters. In particular, this also finally allows for computing global illumination at interactive rates.

**KEYWORDS** 1.3 [Computer Graphics]

## 1 Motivation

Interaktive 3D-Graphik basiert heutzutage fast ausschließlich auf dem Rasterisierungsverfahren, bei dem die einzelnen Dreiecke der Szene sequentiell auf den Bildschirm „gerastert“ werden. Dieses Verfahren ist effizient und günstig in Hardware realisierbar, und daher selbst auf PCs weit verbreitet. Leider stößt dieses Verfahren hinsichtlich Darstellungsqualität und unterstützter

Szenengröße zunehmend an seine Grenzen, da insbesondere indirekte Effekte (wie z. B. Schatten, Reflexionen, usw.) mit diesem Ansatz nur unzureichend simuliert werden können. Diese Limitierungen können zwar in manchen Anwendungen (z. B. in Spielen) oft kaschiert werden – indem z. B. die benötigten Effekte vorberechnet oder mit für jedes Spiel speziell entworfenen Tricks „ge-faked“ werden – sie stellen aber ein zunehmendes Problem insbesondere für Anwendungen wie z. B. Virtual Reality, Virtual Prototyping usw. dar, bei denen solche Tricks und „Fakes“ nur begrenzt anwendbar sind.

Eine Alternative zum Rasterisierungsverfahren ist das Ray Tracing, bei dem – physikalisch

motiviert – virtuelle Lichtstrahlen („rays“) in die Szene zurückverfolgt (ge„traced“) werden. Ist der Schnittpunkt eines Strahles mit der Szenengeometrie erst gefunden, so kann das entlang dieses Strahls transportierte Licht berechnet werden, indem die Beiträge der Lichtquellen berechnet, und indirekte Beiträge (z. B. aus der Reflektionsrichtung) rekursiv mittels „Sekundärstrahlen“ berechnet werden. Da Ray Tracing dem physikalischen Lichttransport nachempfunden ist, liefert es meist sehr realistische Bilder. Durch seine hohen Rechenanforderungen galt es allerdings im Allgemeinen bisher als unvereinbar mit interaktiven Anwendungen.

Um diese Lücke zwischen „unzureichend realistischer Rasterisie-

\*Die Dissertation von Herrn Dr. Wald ist mit dem SaarLB Wissenschaftspreis 2004 ausgezeichnet worden, sowie von der Universität des Saarlandes für den GI-Dissertationspreis 2004 vorgeschlagen worden. Die Gutachter des Promotionsverfahrens waren Prof. Dr.-Ing. Philipp Slusallek, Universität des Saarlandes, Prof. Peter Shirley, University of Utah, und Prof. Philip Dutré, Katholieke Universiteit Leuven.

„einerseits und „offline-only Ray Tracing“ andererseits zu schließen, bieten sich nun zwei Möglichkeiten an – entweder das Ray Tracing Verfahren auf Echtzeit-Performanz zu beschleunigen, oder aber das Rasterisierungsverfahren auf photorealistische Qualität zu erweitern. Dabei wurde bisher im Allgemeinen angenommen, dass eine derartige Beschleunigung des Ray Tracing Verfahrens grundsätzlich nicht realisierbar wäre, sodass fast alle Anstrengungen zur interaktiven hochqualitativen Computergraphik ausschließlich den Ansatz verfolgten, Photorealismus mittels verbesserter Rasterisierungsverfahren erreichen zu wollen. Trotz signifikanter Anstrengungen war dieser Ansatz bisher allerdings nur begrenzt erfolgreich.

## 2 Echtzeit Ray Tracing

Im Gegensatz zu diesem „Standardansatz“ der Computergraphik wurde im Rahmen der Dissertation [1] der eher unkonventionelle Ansatz des „Echtzeit Ray Tracing“ (ERT) verfolgt. Dabei wurden mehrere Aspekte berücksichtigt, insbesondere die Entwicklung eines hochoptimierten ERT-Kerns, ein Framework zum parallelen und skalierbaren Ray Tracing auf PC Clustern, eine Methode zur Unterstützung dynamischer Szenen, sowie eine spezielle ERT-Programmierschnittstelle.

### 2.1 Schneller ERT-Kern

Um einen hochgradig optimierten Ray Tracing Kern zu erhalten, müssen insbesondere die Eigenschaften moderner CPUs beachtet werden: So erreichen moderne CPUs zwar extrem hohe Fließkomma-Performanz, allerdings typischerweise nur, wenn diese durch SIMD-Erweiterungen effizient ausgenutzt wird. Desweiteren sind die langen Pipelines moderner Prozessoren anfällig für komplexen Programmcode, schwer vorhersagbare Speicherzugriffsmuster, und „cache misses“.

Um dies zu berücksichtigen, wurde ein neues – mittlerweile pa-

tentiertes – Verfahren entwickelt, mit dem ganze Pakete von Strahlen gleichzeitig berechnet werden, womit Datenabhängigkeiten minimiert, Latenzen versteckt, Speicherbandbreiten reduziert, und SIMD-Erweiterungen effizient ausgenutzt werden können [1].

Zusätzlich zu diesen „low-level“ Optimierungen wurde darauf geachtet, auch aus algorithmischer Sicht nur die effizientesten Algorithmen zu verwenden. So werden insbesondere die kd-Bäume mit den derzeit besten bekannten Techniken konstruiert, indem erweiterte Kostenabschätzungsfunktionen benutzt werden, um die zu erwartende Anzahl an Traversierungs- und Schnittberechnungsoperationen zu minimieren. In ihrer Gesamtheit erlauben diese Techniken, eine bis zu 30-fach höhere Ausführungsgeschwindigkeit als bei herkömmlichen Ray-Tracern zu erreichen [1], womit interaktive Performanz selbst auf Standard-PCs möglich wird.

### 2.2 Skalierbare Parallelisierung

Da reale Anwendungen oft viele Millionen Strahlen pro Bild benötigen, kann oft selbst ein extrem schneller ERT-Kern nicht immer interaktive Performanz garantieren. Daher wurde das ERT-System so erweitert, dass auf Clustern auch mit mehreren PCs parallel am selben Bild gerechnet werden kann. Dabei musste insbesondere selbst auf Standardnetzen und unter Echtzeitbedingungen eine effiziente Parallelisierung und Skalierbarkeit erreicht werden. Dadurch konnte die Performanz um weitere bis zu zwei Größenordnungen gesteigert, und sogar industrielle VR-Zentren mit der ERT-Technologie betrieben werden<sup>1</sup>.

### 2.3 Dynamische Szenen

Eine der Grundvoraussetzungen für hohe Ray Tracing Performanz ist

<sup>1</sup> Volkswagen betreibt in Wolfsburg eine 3200 × 1200 Pixel „Powerwall“ mit einem 46-Knoten ERT-Cluster zur photorealistischen Fahrzeugvisualisierung.

die Verwendung effizienter Datenstrukturen – wie z. B. kd-Bäumen – zum schnellen Finden des nächsten Schnittpunkts eines Strahls mit der Szene. Diese Datenstrukturen stellen allerdings ein großes Problem für Echtzeit Ray Tracing dar, da sie bei veränderlichen Szenen nur für ein Bild gültig sind, aber meist zu komplex sind, um sie in Echtzeit pro Bild neu aufzubauen.

Für *interaktive* Anwendungen ist es jedoch unerlässlich, auch Interaktionen mit der Szene – also veränderliche Geometrie – zuzulassen. Zu diesem Zweck wurde eine hierarchische Methode entwickelt, in der die Szene in in sich statische „Objekte“ zerlegt wird. Diese werden dann hierarchisch verwaltet, wobei affine Transformationen dadurch ermöglicht werden, dass statt der Objekte die Strahlen transformiert werden, wodurch ein Neubau der Objekte vermieden wird. Durch Neudefinition eines Teils der Objekte pro Frame können auch nichtaffine, freie Bewegungen zumindest zu einem gewissen Grad unterstützt werden.

### 2.4 Out-of-Core Ray Tracing

Ein großer Vorteil des Ray Tracing ist seine nur logarithmische Abhängigkeit von der Szenengröße, sowie dass nur wirklich benötigte Daten auch wirklich angefasst werden. Dadurch ist Ray Tracing sehr gut speziell für komplexe Modelle geeignet. Daher wurden im Rahmen der Dissertation neue Verfahren entwickelt, um auch massivst komplexe CAD Datensätze von mehreren Millionen bis Milliarden von Polygonen in Echtzeit darstellen zu können (siehe Bild 1b). Spezieller Fokus lag hier auf neuen hierarchischen Datenstrukturen, sowie auf speziellen Streaming- und Out-of-core-Verfahren [2], die auch Modelle handhaben können, welche größer als der zur Verfügung stehende Speicher sind<sup>2</sup>.

<sup>2</sup> Das Verfahren wurde unter anderem verwendet, um erstmalig eine komplette, aus 350 Mio. Polygonen bestehende Boeing 777 interaktiv darzustellen.



**Bild 1** Einige praktische Anwendungen des Echtzeit Ray Tracing: a) Visualisierung einer „Hella“ Lampe mit genauer Simulation des Brechungsverhalten im Glaskörper. b) Visualisierung eines extrem komplexen, aus 350 Millionen Dreiecken bestehenden Modells einer Boeing 777. c) Simulation globaler (indirekter) Beleuchtung in einer Architekturszene. d) Interaktive Beleuchtungssimulation in einem Autinnenraum, inklusive indirekter Beleuchtung, Reflektion/Refraktion im Glas, usw. Alle diese Beispiele können interaktiv dargestellt werden, man kann also z. B. durch die Szene „hindurchfliegen“.

## 2.5 Das „OpenRT“ API

Eine Grundvoraussetzung, um das Potential der ERT-Technologie einer breiten Nutzerschicht zu erschließen, sowie deren einfache Verwendung und Erweiterbarkeit zu ermöglichen, ist die Existenz einer geeigneten Programmierschnittstelle (API). Da existierende Graphik-APIs nicht für die ERT-Technologie geeignet waren, musste ein solches API komplett neu entworfen und entwickelt werden. Dieses „OpenRT“ API bildet bereits heute einen Quasi-Standard für die ERT-Technologie.

## 2.6 Entwurf von ERT-Hardware

Basierend auf den Erkenntnissen des Softwarekerns wurde die weltweit erste Hardware-Architektur für Echtzeit Ray Tracing entworfen, indem die im Rahmen der Dissertation entwickelten Konzepte in Spezial-Hardware realisiert wurden. Diese mittlerweile patentierte Hardware wurde bereits prototypisch realisiert, an einer kommerziellen Realisierung wird gearbeitet.

## 3 Interaktive Beleuchtungssimulation

Während beim Ray Tracing schon Effekte wie Schatten, Reflexion und Brechung berücksichtigt werden, verlangt die *globale* Beleuchtungsberechnung zusätzlich noch die Berücksichtigung *indirekter* Beleuchtung, also die mehrfache Interreflektion des Lichtes auch an diffusen Oberflächen. Um solche globalen Effekte zu berechnen, sind eine

Vielzahl von Verfahren bekannt, jedoch ist der Berechnungsaufwand typischerweise nochmals höher als bei „normalem“ Ray Tracing.

Da die meisten dieser Verfahren allerdings auf dem Ray Tracing Verfahren aufbauen, eröffnet das ERT auch ein grosses Potential für schnelle globale Beleuchtungssimulation. Zu diesem Zweck wurden im zweiten Teil der Dissertation mehrere Verfahren entwickelt, die die Vorteile der ERT-Technologie voll ausnutzen, und damit auch die interaktive Globale Beleuchtungssimulation ermöglichen.

So wurde in Kooperation mit der TU Kaiserslautern das „Instant Global Illumination“ Verfahren [3] entwickelt, welches schnelles Ray Tracing, effektive numerische Integration mittels Niederdiskrepanzmustern, ein effizientes Parallelisierungsframework, sowie ein bildbasiertes, kantenerhaltendes Filterverfahren kombiniert, um mittels eines Clusters von PCs interaktive Beleuchtungssimulation zu verwirklichen. Dabei werden so gut wie alle globalen Beleuchtungseffekte – direkte und indirekte Beleuchtung, weiche und harte Schatten, simple Kaustiken, Antialiasing, etc. – unterstützt (Bild 1c).

## 4 Zusammenfassung

Die in der Dissertation vorgestellten Verfahren ermöglichen die Etablierung eines neuen Paradigmas der Computergraphik – des Echtzeit Ray Tracing. Dabei wurden verschiedene Punkte behan-

delt, von neuen schnellen Kernverfahren, über effiziente Parallelisierung, dynamische Szenen, massive Modelle, bis hin zur interaktiven globalen Beleuchtungssimulation. Die entwickelten Techniken werden bereits heute kommerziell von der *inTrace* GmbH vermarktet und von mehreren renommierten Kunden industriell eingesetzt, so z. B. zur Visualisierung kompletter Flugzeuge, Simulation von Scheinwerfern, sowie zur interaktiven photorealistischen Fahrzeugvisualisierung (siehe Bild 1). Für diese innovativen Lösungen wurde die *inTrace* GmbH vor kurzem mit dem European IST Prize ausgezeichnet.

## Danksagung

Viele Personen haben zu den erreichten Forschungsergebnissen signifikant beigetragen, so z. B. Carsten Benthin, Philipp Slusallek, Alexander Keller, Jörg Schmittler, Andreas Dietrich, und viele andere.

## Literatur

- [1] I. Wald. *Realtime Ray Tracing and Interactive Global Illumination*. PhD thesis, Saarland University, 2004.
- [2] I. Wald, A. Dietrich, and P. Slusallek. An Interactive Out of Core Rendering Framework for Visualizing Massively Complex Models. In *Rendering Techniques 2004*.
- [3] I. Wald, T. Kollig, C. Benthin, A. Keller, and P. Slusallek. Interactive Global Illumination using Fast Ray Tracing. *Rendering Techniques*, 2002.



**Dr. Ingo Wald** promovierte nach seinem Informatikstudium an der TU Kaiserslautern 2004 an der Universität des Saarlandes. Seine Dissertation „Realtime Ray Tracing and Interactive Global Illumination“ wurde 2004 mit dem „SaarLB Wissenschaftspreis“ ausgezeichnet, sowie unter anderem für den GI Dissertationspreis 2004 durch

die Universität des Saarlandes nominiert. Nach einem Post-Doc Aufenthalt am Saarbrücker Max-Planck-Institut für Informatik wechselte er Mitte 2005 als Visiting Assitant Professor an die University of Utah. Neben seiner akademischen Laufbahn ist Ingo Wald Mitgründer und technischer Geschäftsführer der *inTrace* Realtime Ray Tracing GmbH, die vor kurzem mit dem „European IST Prize“ ausgezeichnet wurde.

Adresse: SCI Institute, University of Utah, 50 S Central Campus Dr., Salt Lake City, UT, 84112, USA, Tel.: +1 801-587-9455, Fax: +1 801-585-6513, E-Mail: [wald@sci.utah.edu](mailto:wald@sci.utah.edu)