

Efficient Object-Based Hierarchical Radiosity Methods

Stephan Schäfer

Technische Universität Carolo-Wilhelmina zu Braunschweig
Institut für ComputerGraphik
<http://www.cg.cs.tu-bs.de>

Eines der Schwerpunktthemen in der Computergraphik ist die effiziente Erzeugung von fotorealistischen Bildern. Im Gegensatz zur einfachen Bilderzeugung, die bereits durch gängige 3D-Grafikhardware unterstützt wird, gehorcht die fotorealistische Bildsynthese physikalischen Gesetzen, die die Lichtausbreitung innerhalb einer bestimmten Umgebung beschreiben. Durch die Simulation der Energieausbreitung in einer dreidimensionalen Szene können globale Effekte wie Schatten und mehrfache Reflektionen wirklichkeitstreu dargestellt werden.

Die hierarchische Radiositymethode (*Hierarchical Radiosity*) ist eine Möglichkeit, um die globale Beleuchtung innerhalb einer Szene zu berechnen. Da diese Methode auf die Verwendung von rein diffus reflektierenden Oberflächen beschränkt ist, sind damit errechnete Lösungen blickwinkelunabhängig und lassen sich in Echtzeit am Bildschirm durchwandern. Zudem ist dieser Algorithmus aufgrund der verwendeten physikalischen Grundlagen sehr gut zur Beleuchtungssimulation und Architekturvisualisierung geeignet.

Den Schwerpunkt dieser Dissertation stellt die Anwendung objektbasierter Methoden auf das Radiosityproblem dar. Durch konsequente Ausnutzung von Objektinformationen während aller Berechnungsschritte konnten deutliche Verbesserungen im Rahmen der hierarchischen Radiositymethode erzielt werden. Dabei wurden neue Verfahren zur Berechnung der Radiosityfunktion über gekrümmte Oberflächen, zur Parallelisierung dieses Algorithmus sowie zum effizienten Einsatz in der Architekturvisualisierung entwickelt, die im folgenden präsentiert werden.

1 Hierarchical Radiosity auf gekrümmten Oberflächen

Der Hierarchische Radiosity Algorithmus [HS90, HSA91] besitzt eine Zeitkomplexität von $O(k^2 + n)$, wobei k die Anzahl der Eingabepolygone und n die Anzahl der durch den Algorithmus generierten Mesh-Elemente entspricht. In der sogenannten Startphase des Algorithmus werden Verknüpfungen zwischen allen Wurzelknoten der noch zu bildenden Hierarchien angelegt. Dabei lassen sich zwar aufgrund der gegenseitigen Orientierungen bestimmte Verknüpfungen bereits ausschließen, jedoch bleiben $k(k-1)/2$ Paare zu untersuchen, woraus der k^2 -Term resultiert. Daraus wird ersichtlich, daß der Algorithmus nur dann praktikabel einsetzbar ist, wenn $k \ll n$ gilt. Diese Bedingung wird insbesondere bei gekrümmten Oberflächen nicht erfüllt, da sie zur Verwendung durch den Algorithmus zunächst sehr fein tesseliert werden müssen. Die Geometrie muß sogar bereits so stark verfeinert werden, daß alle für die Radiosity-Berechnung notwendigen späteren Unterteilungen bereits vorhanden sind. Dies ist darin begründet, daß der Hierarchical Radiosity Algorithmus seine Eingabepolygone bei Bedarf adaptiv unterteilt um die Radiosity-Funktion durch ein feineres Gitter von finiten Elementen besser approximieren zu können

[CGIB86, HSA91]. Ein Polygon, das zu einem gekrümmten Objekt gehört, läßt sich jedoch nicht sinnvoll unterteilen, da die Krümmungsinformation an dieser speziellen Stelle des Objektes nicht vorhanden ist. Demnach kann das Polygon nur planar unterteilt werden, was voraussetzt, daß die lokale Krümmung des Objektes bereits durch die Eingabepolygone hinreichend gut approximiert wurde. Im Allgemeinen wählt man die Anfangsunterteilung also so, daß gekrümmte Oberflächen nicht mehr unterteilt werden müssen, was der grundlegenden Idee dieses hierarchischen Algorithmus widerspricht.

Als Lösung zu diesem Problem wurde eine objektbasierte Unterteilungsstrategie entworfen, die sowohl die Laufzeit als auch die Güte der Simulation des Hierarchical Radiosity Algorithmus unter Verwendung gekrümmter Oberflächen signifikant verbessert. Voraussetzung dazu ist ein Modellierungs- oder Renderingsystem, welches die Eigenschaften von geometrischen Objekten verwalten kann. Software-Architekturen dieser Art sind nicht nur im Forschungsbereich weit verbreitet (MRT [Fel96], Vision [SS95], MoCaRT++ [GMP96]) sondern bilden auch die Grundlage vieler kommerzieller Produkte (Kinetix 3D Studio MAX, Alias/Wavefront Maya). Jeder unterstützte Objekttyp muß dabei in der Lage sein, den Schnitt zwischen einem Strahl und der Objektoberfläche berechnen zu können. Zudem muß am Schnittpunkt die Oberflächennormale berechenbar sein. Jedes Softwarepaket mit Raytracing-Unterstützung beherrscht diese beiden wichtigen Funktionen, welche die vollständige Beschreibung der Objektgeometrie implizieren.

Die objektbasierte Unterteilungsstrategie verwendet als Eingabe eine möglichst grobe Approximation der gekrümmten Objekte, was in Analogie zur Verwendung planarer Objekte im Hierarchical Radiosity Algorithmus zu sehen ist. Die Unterteilung eines Polygons durch den Algorithmus, dient zur Reduzierung des Fehlers im nächsten Simulationsschritt und wird durch ein sogenanntes Orakel gesteuert [HSA91]. Um einen Fehler zu reduzieren, der durch eine zu grobe Approximation der Geometrie verursacht wurde, können Strahlschnitt und Normalenberechnung wie folgt verwendet werden: Das Polygon wird planar unterteilt (z.B. in Form eines Quadtree) und der oder die neu entstandenen Vertices werden markiert. Von jedem dieser Vertices wird nun ein Strahl in Richtung der Oberfläche des Objektes geschickt. Die Position des entsprechenden Vertex wird dann mit dem gefundenen Schnittpunkt gleichgesetzt, wodurch der Vertex auf die Oberfläche des gekrümmten Objektes verschoben wird. Falls das zugehörige Objekt nicht konvex ist, muß die Strahlrichtung umgekehrt werden, sofern die Eingangstesselierung dadurch entstanden ist, daß Punkte von der Oberfläche des Objektes zu Polygonen verbunden wurden. Diese Funktionalität läßt sich generisch zur Verfügung stellen, da durch Vererbungsmechanismen der objektorientierten Programmierung, die objektspezifischen Funktionen automatisch ausgewählt werden. Zudem gibt es die Möglichkeit der gezielten Optimierung für ausgewählte Objektklassen. Beispielsweise verfügen Freiformflächen über eine Parametrisierung, mit der kostengünstiger als durch Strahlschuß Oberflächenpunkte generiert werden können.

Um zu gewährleisten, daß die Energiebilanz innerhalb einer Unterteilungshierarchie korrekt ist, verwendet der Hierarchical Radiosity Algorithmus eine sogenannte *push/pull*-Funktion. Energie, die an verschiedenen Hierarchieknoten eingesammelt wurde, wird anteilsweise im Baum propagiert. Da Radiosity in Energie pro Fläche gemessen wird, er-

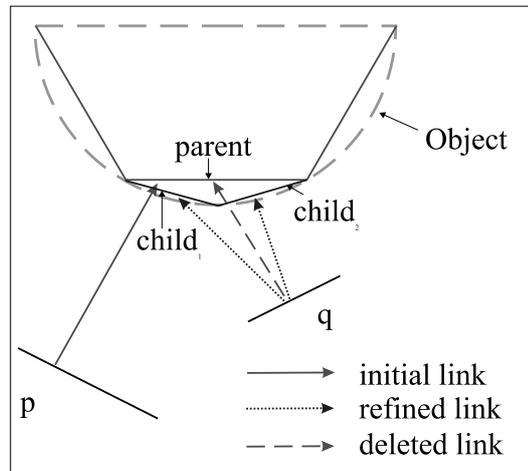


Abbildung 1: Geometrische Verhältnisse der objektbasierten Unterteilung. Dargestellt ist der Schnitt durch das Mesh einer Kugel, deren Ausdehnung durch die gestrichelte Linie angedeutet wird.

halten tieferliegende Knoten denselben Wert wie ihr Vaterknoten, höherliegende Knoten hingegen den nach Flächenanteil gewichteten Wert ihrer Kindknoten. Die objektbasierte Unterteilungsstrategie verletzt jedoch die implizit gemachte Voraussetzung, daß sich die Geometrie eines Knotens direkt aus der Geometrie der Kindknoten zusammensetzt. Durch die Anpassung der Vertices nach einer Unterteilung, stimmt die Orientierung der Fläche im Allgemeinen nicht mehr mit der der Ursprungsfläche überein. Der Anteil der Radiosity eines Kindknotens darf also nicht nach dem Flächenanteil gewichtet werden, sondern nach dem Anteil der in Richtung des Vaterknotens *projizierten* Fläche. Abbildung 1 verdeutlicht diesen Zusammenhang. Die projizierte Fläche berechnet sich durch Multiplikation der Fläche mit dem Kosinus des Winkels zwischen Oberflächennormale und der Projektionsrichtung, in diesem Falle der Normale des Vaterknotens.

Zur exakten Bestimmung von Formfaktoren und Sichtbarkeit zwischen gekrümmten Objekten reicht deren insbesondere zu Beginn des Verfahrens noch sehr grobe Approximation nicht aus. Daraus resultierende Schatten würden deutliche Artefakte aufweisen, da sehr viele Schattenstrahlen ein verdeckendes Objekt nie treffen würden. In beiden Fällen (Formfaktor und Sichtbarkeit) wird stattdessen auf das Originalobjekt zurückgegriffen. Die Sichtbarkeitsberechnungen mittels Strahlschuß werden dadurch genauer und effizienter. Da weniger Objekte auf Sichtbarkeit zu überprüfen sind, kann trotz teurerer Schnittberechnungen mit komplexen Objekten ein Geschwindigkeitsvorteil gegenüber den Tests mit sehr vielen approximierenden Polygonen erzielt werden. Da in die Formfaktorberechnung der Winkel zwischen Oberflächennormale und Transportrichtung eingeht, ermöglicht das entwickelte Verfahren zudem eine sehr gute Näherungsbestimmung des Formfaktorintegrals mittels Monte-Carlo sampling [CW93]. Die Oberflächennormalen werden dazu punktgenau auf der echten Objektoberfläche bestimmt (diese Funktionalität war Voraus-

setzung) und nicht durch die Normale eines Polygons gemittelt.

Zur Rekonstruktion der Radiosity-Funktion auf gekrümmten Oberflächen können sowohl durch Hardware unterstützte Verfahren als auch strahlschußbasierte Verfahren eingesetzt werden. Die Anzeige mittels Grafikhardware verlangt dazu eine gleichmäßige Approximation der Objekte durch planare Polygone. Je nach eingesetztem Orakel erzeugt die objektbasierte Unterteilungsstrategie sehr unregelmäßige Polygongitter. Das häufig verwendete BF- oder BFA-refinement [HSA91, Sil95], welches den Simulationsfehler mit der Größe der Radiosity-Funktion gleichsetzt, zeigt dieses Verhalten sehr deutlich wenn Teile eines Objektes im Schatten liegen während anderer Bereiche direkter Beleuchtung ausgesetzt sind. Durch Verwendung eines Orakels welches Fehlerschranken bei allen Komponenten des Energietransports berücksichtigt [LSG94, GH96, SSS97], können ausgewogenere Unterteilungen generiert werden. Dazu muß bei der Berechnung der Fehlerschranken zusätzlich die lokale Krümmung berücksichtigt werden. Durch Messen der Abweichung der zu einem Gitterelemente gehörenden Normalenvektoren untereinander, kann eine Aussage über die Güte der Krümmungsapproximation gemacht werden. Nach Vergleich mit einem Schwellwert können dann weitere Unterteilungen angefordert werden. Der gleiche Test kann auch dazu verwendet werden, um *nach* der Radiosity-Berechnung das Gitter gekrümmter Objekte lokal nachzubessern. Da dies im Allgemeinen in Regionen eines geringen Simulationsfehlers nötig ist, können die vorhandenen Radiosity-Werte ohne Neuberechnung übernommen werden. Dies führt mit sehr geringem Rechenaufwand zu Radiosity-Lösungen, die durch Ausnutzung von Grafikhardware mit hoher Qualität angezeigt werden können. Abbildung 2 zeigt die Anwendung des Algorithmus auf eine künstliche Szene die verschiedene gekrümmte Objekte beinhaltet.

Für die Generierung von hochaufgelösten Einzelbildern läßt sich durch erhöhten Rechenaufwand die Qualität noch deutlich verbessern. Dazu wird ein *final-gathering* durchgeführt, welches wiederum durch Ausnutzung von Objektinformationen an obiges Verfahren angebunden werden kann. Zur Berechnung der Formfaktoren von den sichtbaren Oberflächenpunkten zum Auge, kann der gleiche Formfaktoralgorithmus verwendet werden. Gekrümmte Objekte können dadurch *exakt* abgebildet werden, da, wie im Raytracing-Verfahren, jeder sichtbare Oberflächenpunkt berücksichtigt wird. Dadurch können auch weitere Shadingverfahren wie z.B. *bump-mapping* durch lokale Manipulation der Normalen direkt integriert werden. Von den Verbesserungen des vorgestellten Radiosity-Algorithmus profitiert das objektbasierte *final-gathering* dabei ganz erheblich. Bereits nach sehr wenigen Iterationen können hochqualitative Ausgaben berechnet werden, da kein fein aufgelöstes Gitter für eine polygonbasierte Anzeige benötigt wird. Da gerade bei der Verwendung von Szenen mit gekrümmten Objekten die Startphase des Hierarchical Radiosity Algorithmus deutlich verkürzt werden konnte, ist die Kombination beider Verfahren bei diesen Szenen äußerst effizient.



Abbildung 2: Radiosity-Lösung einer Szene mit verschiedenen gekrümmten Objekten unter Verwendung der objektbasierten Unterteilungsstrategie.

2 Distributed Hierarchical Radiosity

Aufbauend auf den Ergebnissen des Radiosity-Algorithmus für gekrümmte Oberflächen kann eine effiziente Parallelisierung des Hierarchical Radiosity Algorithmus implementiert werden. Durch Ausnutzung von Objektinformationen läßt sich eine verteilte Berechnung mit sehr geringem Kommunikationsaufwand realisieren, die, kombiniert mit einem adaptiven Schedulingverfahren, gut skalierbar ist. Bisherige Ansätze zur Parallelisierung des Hierarchical Radiosity Algorithmus [BG95, Fun96] erzielen zwar sehr gute Ergebnisse, abstrahieren aber nicht weiter von den Eingabepolygonen. Die Begrenzung der Kommunikationskosten stellt daher einen nicht unerheblichen Anteil an der Implementierung dar.

Der größte Berechnungsaufwand wird durch die Bestimmung der Formfaktoren oder genauer durch die Bestimmung der gegenseitigen Sichtbarkeit zweier Hierarchieknoten verursacht. Es sollte also genau dieser Teil des Algorithmus auf verschiedene Prozessoren verteilt werden. Unter Verwendung der Szenenobjekte für die Formfaktor- und Sichtbarkeitsberechnung ist es damit möglich, nur sehr geringe Datenmengen im Netz verteilen zu müssen. Ausgehend von einer *Client-Server*-Architektur wird der Algorithmus so aufgeteilt, daß der Server den speicherintensiven aber rechenzeitarmen Energietransport berechnet, während die Klienten Formfaktoren und Sichtbarkeit bestimmen. Diese Aufgaben benötigen die größte Rechenzeit und können aufgrund des Objektmodells mit sehr wenig zusätzlichem Speicheraufwand realisiert werden.

	# of clients							
	1	2	3	4	5	6	7	8
total time [s]	32695	16548	11402	8373	7550	5676	5215	4650
speed-up	1.0	1.98	2.9	3.9	4.3	5.8	6.3	7.0
max. comm. time [%]	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
max. idle. time [%]	0.4	2.9	3.2	3.3	4.9	4.2	6.8	9.4

Abbildung 3: Skalierbarkeit des verteilten hierarchischen Radiosity-Algorithmus (ohne Clustering) für bis zu 8 Klienten. Die Messungen erfolgten auf identischen SGI Indy Workstations.

Das klassische Link-Refinement berechnet nach Erzeugen eines neuen Links den zugehörigen Formfaktor und vermerkt ihn in der Datenstruktur. Zur effizienten Parallelisierung wird dieser Schritt nun verzögert ausgeführt. Alle Links werden zunächst ohne Formfaktor angelegt. Nach Abschluß der Refinementprozedur werden alle zu berechnenden Formfaktoren auf die vorhandenen Prozessoren verteilt und dort berechnet. Die zu dieser Berechnung notwendigen Daten werden so kompakt wie möglich übertragen und bestehen im Wesentlichen aus den Bezeichnern der beteiligten Objekte, zweier Referenzpunkte und der Ausdehnung der auf dem Server verwendeten Patches. Dies erlaubt die Berechnung der *point-to-disc*-Formfaktoren [CW93] auf den Klienten ohne Verwendung eines finitelemente Meshes. Die Klienten greifen ausschließlich auf die Originalobjekte zurück und verwenden deren Raytracing-Fähigkeiten, d.h. Strahlschnitt und Berechnung der Oberflächennormalen (s.o.). Dies begründet den geringen Speicherbedarf der Klienten, da das Radiosity-Mesh dort nicht gespeichert und insbesondere dessen Veränderungen nie übertragen werden müssen. Nach Rückübertragung aller Ergebnisse zum Server wird dort der Energietransport berechnet und die nächste Verfeinerung gestartet.

Bei der Entwicklung des Algorithmus wurde auf dessen Verwendbarkeit in realen Umgebungen, d.h. einem Netzwerk von Workstations, Wert gelegt. Dazu wurde ein load-balancing implementiert, welches je nach Leistungstärke und Rechenlast der Klienten die Größe der Formfaktorpakete anpaßt. Ebenso wurde die Möglichkeit vorgesehen, daß bei Unterbeschäftigung von Klienten Pakete entzogen oder umverteilt werden können. Abbildung 3 demonstriert die dadurch erzielbare Skalierung anhand einer Beispielszene von ca. 2000 Polygonen ohne den Einsatz weiterer Beschleunigungstechniken. Der geringe Anteil der Kommunikationskosten an der Gesamtrechenzeit ist ebenfalls ersichtlich.

3 Clustering-Strategie für Hierarchical Radiosity

Die Verwendung von Hierarchien oberhalb der Eingabepolygone stellt die logische Fortsetzung des Hierarchical Radiosity Algorithmus dar und erlaubt die Berechnung von Radiosity-Lösungen für sehr große Szenen [Sil95, SAG94]. Dabei werden Eingabepolygone aufgrund ihrer räumlichen Nachbarschaft gruppiert (*clustering*) und Hierarchien, z.B. *bounding-volume* Hierarchien oder $k - d$ Bäume gebildet, zwischen deren Ebenen der Energieaustausch repräsentativ für die darunterliegende Geometrie berechnet werden kann. Analog

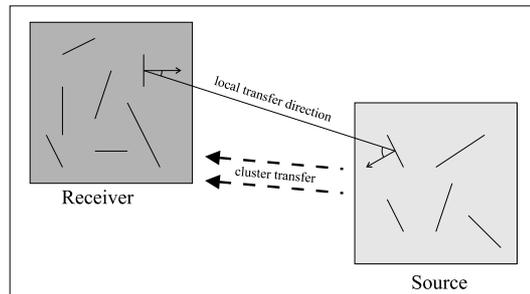


Abbildung 4: Die gegenseitigen Orientierungen der beteiligten Flächen müssen beim Energietransport zwischen Clustern berücksichtigt werden.

zum Hierarchical Radiosity Algorithmus wird der Energietransport dort auf niedrigeren Ebenen berechnet, wo der lokale Fehler einen Grenzwert überschreitet. Die Verwendung von Links zwischen Clustern erlaubt die Reduktion der quadratischen Startphase des Hierarchical Radiosity Algorithmus auf eine einzelne Operation, nämlich der Erzeugung eines Links der gesamten Szene auf sich selbst und anschließender adaptiver Verfeinerung dieser Links.

Zur Berechnung des Energietransportes zwischen Clustern verwendet Sillion ein Volumenmodell gemeinsam mit der Annahme, daß Flächen, die zu einem Volumen oder Cluster zusammengefaßt wurden, das Licht isotropisch streuen [Sil95]. Diese Vereinfachung erlaubt die Definition von einheitlichen Formfaktoren zwischen allen Kombinationen aus Volumina und Flächen, wobei die Sichtbarkeitsberechnung für Volumina effizient über eine Dämpfungsfunktion realisiert werden kann.

Sowohl die Qualität der Radiosity-Lösung als auch die benötigte Laufzeit hängen sehr stark vom Aufbau der verwendeten Hierarchie ab [HDS99]. Die Hierarchie muß den Energietransport bestmöglich unterstützen, da der Algorithmus von der Annahme ausgeht, daß auf höheren Leveln der Energietransport niedrigerer Level approximiert werden kann. Für Sichtbarkeitsberechnungen zwischen Flächen ist obiges Volumenmodell zu ungenau, weswegen strahlschußbasierte Sichtbarkeitstest verwendet werden. Da sich Strahlschußverfahren sehr gut durch hierarchische Datenstrukturen beschleunigen lassen, sollte die verwendete Hierarchie auch dazu geeignet sein. Eine zu starke Verästelung der Hierarchie führt bei der Unterteilung von Links dazu, daß für sehr viele Nachfahren neue Links angelegt werden müssen, was wiederum eine negative Auswirkung auf die Gesamtlaufzeit hat. Die Hierarchie sollte daher einen geringen Verzweigungsgrad aufweisen. Schließlich sollte die Zeit zur Erzeugung der Hierarchie deutlich unter der Laufzeit des Radiosity-Algorithmus liegen.

Zur Berechnung hochqualitativer Radiosity-Lösungen sehr großer Szenen, wurde im Rahmen dieses Projektes eine Clustering-Strategie angewendet, die innerhalb der Forschungsgruppe entwickelt wurde und zur Beschleunigung von Raytracingalgorithmen dient [MF99]. Die durch den Algorithmus aufgebaute *bounding-volume* Hierarchie verwendet in ihren



Abbildung 5: Lichtsimulation in einer Architekturszene mittels Radiosity-Clustering. Das gesamte Gebäude wurde mit 90 000 Polygonen modelliert.

inneren Knoten Datenstrukturen zur regulären Raumunterteilung, falls die entsprechende Region räumlich gleichmäßig besetzt ist. Zur Steuerung der Qualität der Hierarchie wird eine Kostenfunktion verwendet, die die Unterteilung eines Knotens aufgrund der Anzahl und der Oberflächengröße der enthaltenen Objekte vornimmt. Die Kombination dieser Techniken führt zu einer *bounding-volume* Hierarchie, die Objektdetails sehr gut erhält und dank ihrer Hybridstruktur Strahlschnittanfragen sehr schnell beantworten kann. Die zugrundeliegende Datenstruktur der Hierarchie ist ein Binärbaum. Dies garantiert minimale Kosten bei der Verfeinerung von Links durch den Hierarchical Radiosity Algorithmus. Die dadurch bedingten sehr tiefen Hierarchien verursachen jedoch erhöhte Traversierungskosten, die beim Energietransfer zwischen Clustern sowie in der *push/pull*-Funktion (siehe Abschnitt 1) auftreten. Der Energietransfer zwischen Clustern wird zur Erhöhung der Qualität hier nicht mithilfe des Modells der isotropisch streuenden Volumina bestimmt. Stattdessen wird die Orientierung aller beteiligten Flächen berücksichtigt, was für jede Interaktion eine Traversierung der entsprechenden Teilhierarchie verursacht. Abbildung 4 verdeutlicht die Notwendigkeit dieser Methode.

Diese Vorgehensweise ähnelt den in [SAG94] verwendeten α -links. Dabei werden die über einen Cluster-Link empfangenen Radiositäten jedoch direkt (und wie auch in [GH96]) unter Berücksichtigung der lokalen Einfallsrichtung auf die enthaltenen Flächen projiziert. Durch Verwendung eines Stacks läßt sich der erforderliche Traversierungsaufwand mi-

nimieren. Dabei wird zunächst der sendende Cluster traversiert und die beinhalteten Radiosities und Orientierungen zwischengespeichert. Der Inhalt des Stacks wird dann direkt auf die Flächen des empfangenden Clusters angewandt, so daß pro Energietransfer zwischen Clustern nur 2 Traversierungen benötigt werden. Zur weiteren Optimierung wird die Radiosity-Lösung im Gauss-Seidel Iterationsverfahren berechnet, welches schneller konvergiert als die üblicherweise für Hierarchical Radiosity verwendete Jacobi-Iteration [LSG94]. Dabei werden die Radiosity-Werte ohne Zwischenwertspeicherung aktualisiert was weitere Hierarchie-Traversierungen vermeidet.

Die Anwendung der beschriebenen Techniken führte zu einem Clustering-Algorithmus für Hierarchical Radiosity, der sehr laufzeit- und speichereffizient arbeitet. Dieselbe Datenstruktur konnte dabei erfolgreich zur Simulation des Energietransports als auch zur schnellen Bestimmung von Sichtbarkeitsverhältnissen genutzt werden. Abbildung 5 zeigt eine reale Architekturszene, auf die der entwickelte Algorithmus angewandt wurde.

Literaturverzeichnis

- [BG95] Bohn, C.-A.; Garmann, R.: A Parallel Approach to Hierarchical Radiosity. In Proceedings of the Winter School of Computer Graphics and CAD Systems '95 (Skala, V., Hg.). University of West Bohemia, Plzen, Czech Republic, February 1995, S. 26–35.
- [CGIB86] Cohen, M.; Greenberg, D. P.; Immel, D. S.; Brock, P. J.: An Efficient Radiosity Approach for Realistic Image Synthesis. In IEEE Computer Graphics and Applications, Bd. 6 (3):(1986), S. 26–35.
- [CW93] Cohen, M. F.; Wallace, J. R.: Radiosity and Realistic Image Synthesis. Academic Press Professional, Boston, MA, 1993.
- [Fel96] Fellner, D. W.: MRT – A Teaching and Research Platform for 3D Image Synthesis. In IEEE CG&A, Bd. 16 (3).
- [Fun96] Funkhouser, T. A.: Coarse-Grained Parallelism for Hierarchical Radiosity Using Group Iterative Methods. In Computer Graphics Proceedings, Annual Conference Series, 1996 (ACM SIGGRAPH '96 Proceedings), 1996, S. 343–352.
- [GH96] Gibson, S.; Hubbard, R. J.: Efficient Hierarchical Refinement and Clustering for Radiosity in Complex Environments. In Computer Graphics Forum, Bd. 15 (5):(1996), S. 297–310.
- [GMP96] Garmann, R.; Mencl, R.; Pietrek, G.: MoCaRT++: An Object Oriented System for Monte Carlo Image Synthesis. Techn. Ber. RR 618, Universität Dortmund, Dortmund, Germany, June 1996.
- [HDS99] Hasenfratz, J. M.; Domez, C.; Sillion, F.; Drettakis, G.: A Practical Analysis of Clustering Strategies for Hierarchical Radiosity. In Computer Graphics Forum (Proc. Eurographics '99), September 1999, Bd. 18, S. C–221–C–232.
- [HS90] Hanrahan, P.; Salzman, D.: A Rapid Hierarchical Radiosity Algorithm for Unoccluded Environments. Technical Report CS-TR-281-90, Department of Computer Science, Princeton University, Princeton, NJ, August 1990.

- [HSA91] Hanrahan, P.; Salzman, D.; Aupperle, L.: A Rapid Hierarchical Radiosity Algorithm. In Computer Graphics (ACM SIGGRAPH '91 Proceedings), July 1991, Bd. 25, S. 197–206.
- [LSG94] Lischinski, D.; Smits, B.; Greenberg, D. P.: Bounds and Error Estimates for Radiosity. In Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings), 1994, S. 67–74.
- [MF99] Müller, G.; Fellner, D. W.: Hybrid Scene Structuring with Application to Ray Tracing. In Proceedings of the International Conference on Visual Computing (ICVC'99), Goa, India, February 1999, S. 19–26. Available online via <http://www.cg.cs.tu-bs.de/people/mueller/publications>.
- [SAG94] Smits, B.; Arvo, J.; Greenberg, D.: A Clustering Algorithm for Radiosity in Complex Environments. In Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings), 1994, S. 435–442.
- [Sil95] Sillion, F.: A Unified Hierarchical Algorithm for Global Illumination with Scattering Volumes and Object Clusters. In IEEE Transactions on Visualization and Computer Graphics, Bd. 1 (3).
- [SS95] Slusallek, P.; Seidel, H.-P.: Vision - An Architecture for Global Illumination Calculations. In IEEE Transactions on Visualization and Computer Graphics, Bd. 1 (1):(1995), S. 77–96. Available from <http://www9.informatik.uni-erlangen.de/eng/research/pub95>.
- [SSS97] Stamminger, M.; Slusallek, P.; Seidel, H.-P.: Bounded Radiosity - Illumination on General Surfaces and Clusters. In Computer Graphics Forum (Eurographics '97 Proceedings), Bd. 16 (3):(1997), S. C309–C317. Available from <http://www9.informatik.uni-erlangen.de/eng/research/pub1997>.

Stephan Schäfer, geboren am 28. November 1965 in Bonn, Abitur 1985 am Collegium Josephinum Bonn, Diplom in Informatik 1993 an der Rheinischen Friedrich-Wilhelms-Universität Bonn, Promotion in Informatik 2000 an der Technischen Universität Carolo-Wilhelmina zu Braunschweig. Seit Juli 2000 als Softwarespezialist bei SUN microsystems, Hamburg.