

Wavelet-basierte globale Beleuchtungsalgorithmen

Diplomarbeit, vorgelegt von Frank Büllersfeld

Rheinische Friedrich-Wilhelms-Universität Bonn
Institut für Informatik III

22. Juni 1999

Versicherung

Hiermit versichere ich an Eides Statt, daß ich die vorliegende Arbeit „Wavelet-basierte globale Beleuchtungsalgorithmen“ selbständig verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Bonn, den 22. Juni 1999

Frank Bülesfeld

Danksagung

Ich möchte allen herzlich danken, die mich bei der Anfertigung dieser Arbeit unterstützt haben.

Herrn Professor Dr. D. W. Fellner danke ich für die – trotz des zwischenzeitlich erfolgten Umzugs der von ihm geleiteten ComputerGraphik-Gruppe von Bonn nach Braunschweig – ausgezeichneten Rahmenbedingungen für das Anfertigen dieser Arbeit.

Stephan Schäfer danke ich dafür, daß ich mich mit einem Teilproblem seines sehr interessanten Projekts habe beschäftigen dürfen. Der praktische Teil dieser Arbeit profitiert entscheidend von seiner „Pionierarbeit“ in Sachen Anbindung von Finite-Elemente-Methoden zum approximativen Lösen des globalen Beleuchtungsproblems an das MRT. Trotz eigenen Termindrucks hatte er stets ein offenes Ohr für Fragen und Probleme.

Gordon Müller habe ich es zu verdanken, daß seinerzeit ein enger Kontakt zur Computer-Graphik-Gruppe zustande gekommen ist.

Annette Scheel (Universität Erlangen-Nürnberg) danke ich für ihr freundliches Entgegenkommen, mir Einsicht in ihre beeindruckende Diplomarbeit zum Thema „Physikalisch basierte Beleuchtungsrechnung mit Wavelet-Radiance“ zu gewähren. Leider konnte der praktische Teil dieser Arbeit von dem Erkenntnisgewinn nicht mehr profitieren, da das Fundament der Implementierung zu diesem Zeitpunkt bereits gelegt war.

Schließlich möchte ich mich bei Beate und Bernd Eyermann sowie bei meinem Bruder Lutz für das detaillierte Korrekturlesen bedanken.

Inhaltsverzeichnis

1	Einleitung	6
2	Theoretischer Hintergrund	9
2.1	Physikalisch basiertes Beleuchtungsmodell	9
2.1.1	Abstraktionen	10
2.1.2	Raumwinkel	10
2.1.3	Radiometrische Größen	11
2.1.4	BRDF	12
2.1.5	Radiance-Gleichung	13
2.2	Mathematische Werkzeuge	15
2.2.1	Hilbert-Raum $\mathcal{L}^2(D)$	16
2.2.2	Galerkin-Methode	16
2.2.3	Gauss-Seidel-Verfahren	19
2.2.4	Wavelet-Basen	20
2.2.4.1	Motivation	20
2.2.4.2	Eindimensionale Haar-Basis	21
2.2.4.3	Mehrdimensionale Haar-Basis	25
2.2.4.4	Standard-Zerlegung des Transportoperators	27
2.2.4.5	Auswahl einer Wavelet-Basis	28
2.3	Basis-Algorithmus	29
3	Wavelet-Radiosity	31
3.1	Implementierung des Basis-Algorithmus	31
3.1.1	Datenstrukturen	31
3.1.1.1	Verwaltung der Radiance-Koeffizienten	31
3.1.1.2	Verwaltung der Transportkoeffizienten	32
3.1.2	Berechnung der Transportkoeffizienten	34
3.1.3	Lösen der diskreten Radiance-Gleichungen	36
3.1.4	Verfeinerung der diskreten Radiance-Gleichungen	36
3.1.4.1	Orakel	37
3.1.4.2	Adaptive numerische Integration	38
3.2	Vergleich mit dem Hierarchical-Radiosity-Algorithmus	39
3.3	Minimal Rendering Toolkit	40
3.3.1	Einführung	40
3.3.2	Anbindung des Basis-Algorithmus	41
3.4	Erweiterungen	43
3.4.1	Gekrümmte Objekte	43
3.4.2	Final-Gathering	43
3.4.2.1	Basis-Algorithmus	43

3.4.2.2	Beschleunigungs-Techniken	46
3.4.2.3	Gekrümmte Objekte	47
3.5	Ergebnisse	48
4	Wavelet-Radiance	52
4.1	Implementierung des Basis-Algorithmus	52
4.1.1	Datenstrukturen	52
4.1.1.1	Verwaltung der Radiance-Koeffizienten	52
4.1.1.2	Verwaltung der Transportkoeffizienten	54
4.1.2	Berechnung der Transportkoeffizienten	54
4.1.3	Lösen der diskreten Radiance-Gleichungen	56
4.1.4	Verfeinerung der diskreten Radiance-Gleichungen	56
4.1.5	Zusätzliche Aspekte	57
4.1.5.1	Reflexionsmodell	57
4.1.5.2	Partitionierung von Richtungsträgern	57
4.1.5.3	Extraktion einer sichtabhängigen Lösung	59
4.1.6	Anbindung an das MRT	60
4.2	Erweiterungen	60
4.2.1	Sonderbehandlung von diffusen Reflektoren	60
4.2.2	Final-Gathering	61
4.3	Ergebnisse	61
5	Zusammenfassung und Ausblick	65
A	Programm-Optionen	68
	Literaturverzeichnis	71

Kapitel 1

Einleitung

Ein klassisches Problem in der Computergraphik ist die *photorealistische Bild-Synthese*. Gegeben sei eine statische synthetische Szene bestehend aus einer Menge von dreidimensionalen geometrischen Objekten, von denen einige (i. a. wenige) als Lichtquellen ausgezeichnet sind. Für alle Objekte seien wenigstens die Reflexionseigenschaften ihrer Oberflächen bekannt und für die Lichtquellen zusätzlich die Emissionseigenschaften. Ferner seien die Sichtparameter vorgegeben. Ziel ist die Berechnung eines Bildes, das von einer Photographie der korrespondierenden realen Szene nicht zu unterscheiden ist.

Ein neben der Modellierung der Szenengeometrie wesentliches Teilproblem ist das *globale Beleuchtungsproblem*. Gesucht ist die Lichtverteilung über einer Menge von Oberflächenpunkten, die sich nach dem „Einschalten“ der Lichtquellen einstellt. Die Lösungsidee besteht darin, die komplexen physikalischen Phänomene betreffend Ausbreitung und Reflexion von Licht zu *simulieren*. Global bedeutet in diesem Zusammenhang, daß die gesamte Umgebung in die Beleuchtungsrechnung einbezogen wird. Bei der Berechnung des von einem Oberflächenpunkt in eine Richtung ausstrahlenden Lichts wird neben der direkten Beleuchtung durch Emission der Lichtquellen auch die *indirekte Beleuchtung* durch (ggf. mehrfache) Reflexion an der Umgebung berücksichtigt. Außerdem ist die Abschwächung des Lichtflusses durch blockierende Objekte Bestandteil des Kalküls.

Das Spektrum der Reflexionseigenschaften realer Oberflächen reicht von *diffus* bis *glossy* (vgl. Abbildung 1.1). Bei der diffusen Reflexion wird das einstrahlende Licht gleichmäßig in alle möglichen Richtungen reflektiert. Der diffuse Anteil dominiert z. B. bei matten Oberflächen. Bei der glossy Reflexion wird das einstrahlende Licht vorwiegend in einen schmalen Richtungs-Kegel um die gespiegelte Einstrahlrichtung reflektiert. Der glossy Anteil dominiert z. B. bei polierten, lackierten oder nassen Oberflächen. Im *Radiosity-Fall* enthält die Szene ausschließlich diffus emittierende und diffus reflektierende Objekte. Im allgemeinen *Radiance-Fall* ist die Einschränkung hinsichtlich der Reflexionseigenschaften aufgehoben. Die Szene darf insbesondere auch glossy reflektierende Objekte enthalten.

Die meisten Ansätze zur approximativen Lösung des globalen Beleuchtungsproblems lassen sich einer der beiden folgenden Kategorien zuordnen: *Monte-Carlo* (MC) und *Finite-Elemente* (FE). MC-Methoden berechnen die Lichtverteilung über einer Menge von *sichtbaren* Oberflächenpunkten *in Richtung Augpunkt* stochastisch mittels Strahlverfolgung. Dabei wird der Transport einzelner Photonen von den Lichtquellen durch die Szene zu dem Augpunkt simuliert. FE-Methoden berechnen demgegenüber die Lichtverteilung grundsätzlich über *allen* Oberflächenpunkten in *alle* Richtungen bzgl. einer geeigneten endlich-dimensionalen Basis mittels Lösen eines Linearen Gleichungssystems. FE-Methoden haben gegenüber MC-Methoden den konzeptionellen Vorteil, daß sie – auf Kosten eines nicht zu vernachlässigenden Speicherbedarfs – eine *sichtunabhängige* Lösung berechnen.

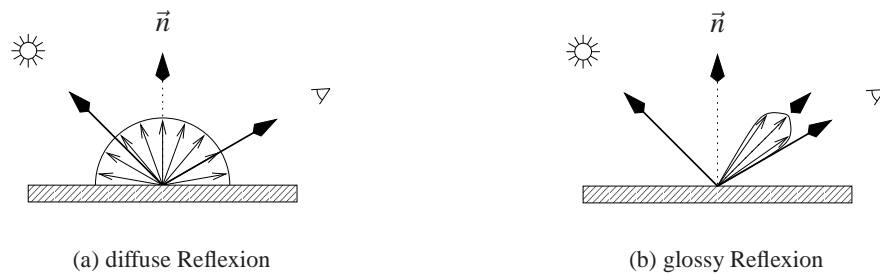


Abbildung 1.1 Reflexionseigenschaften

FE-Methoden für den Radiosity-Fall sind in den vergangenen Jahren kontinuierlich weiterentwickelt worden. Meilensteine bei der Weiterentwicklung waren der *Hierarchical-Radiosity-Algorithmus* [HSA91] und *Clustering-Algorithmen* [SAG94]. Der Hierarchical-Radiosity-Algorithmus läuft formal auf die Verwendung von einfachen Wavelet-Basen zur Approximation der Lichtverteilung hinaus. Wavelet-Basen ermöglichen neben einer kompakten adaptiven Darstellung der Lichtverteilung insbesondere den effizienten Transport von Licht auf einem geeigneten *level-of-detail*. Clustering-Algorithmen erweitern diesen hierarchischen Ansatz in bottom-up Richtung, indem lokale Ansammlungen von Objekten beim Lichttransport ggf. als Einheit betrachtet werden. Derart ausgefeilte FE-Methoden erlauben das globale Beleuchten von komplexen diffusen Szenen, die hardwareunterstützt interaktiv durchwandert werden können.

Die Erweiterung von FE-Methoden für den allgemeinen Radiance-Fall ist Gegenstand aktueller Forschung [Chr95] [SSG⁺99]. Die gesuchte Lichtverteilung hat im Radiance-Fall eine höhere Dimensionalität. Sie variiert nicht nur über den Oberflächenpunkten, sondern i. a. auch über den Ausstrahlrichtungen. Während die Lichtverteilung über einem vorgegebenen Oberflächenpunkt eines diffus reflektierenden Objekts konstant ist, ist die Lichtverteilung über einem vorgegebenen Oberflächenpunkt eines glossy reflektierenden Objekts eine (ggf. komplexe) Funktion über der Menge aller Ausstrahlrichtungen. Diffus reflektierende Oberflächen sehen aus allen Blickrichtungen gleich aus. Demgegenüber ändert sich z. B. die Lage von Glanzlichtern auf glossy reflektierenden Oberflächen mit der Blickrichtung. Zeit- und Speicheraufwand der FE-Methoden nehmen im Vergleich zum Radiosity-Fall deutlich zu.

Ziel dieser Arbeit war die Implementierung einer vielversprechenden Wavelet-basierten FE-Methode für den allgemeinen Radiance-Fall auf Basis des 3D-Graphiksystems *Minimal Rendering Toolkit* (MRT). Der von Christensen vorgeschlagene und als Grundlage dieser Arbeit ausgewählte Wavelet-Radiance-Algorithmus [Chr95] basiert auf der ungewöhnlichen Standard-Zerlegung des Transportoperators, wobei die Lichtverteilung explizit bzgl. Nicht-Standard-Haar-Basen repräsentiert wird. Die Implementierung erfolgte in zwei Phasen. Zunächst wurde ein Wavelet-Radiosity-Algorithmus implementiert, der ebenfalls auf der Standard-Zerlegung des Transportoperators basiert. Danach wurde darauf aufbauend der Wavelet-Radiance-Algorithmus realisiert.

Im nächsten Kapitel wird der theoretische Hintergrund von Wavelet-basierten globalen Beleuchtungsalgorithmen aufgearbeitet. Zunächst wird das globale Beleuchtungsproblem durch ein physikalisch basiertes Beleuchtungsmodell formalisiert. Es folgt eine mathematisch orientierte Beschreibung von Wavelet-basierten FE-Methoden zum approximativen Lösen des globalen Beleuchtungsproblems.

Im Kapitel 3 wird die Implementierung des Wavelet-Radiosity-Algorithmus beschrieben. Die Integration von gekrümmten Objekten erfolgt analog zu der bereits existierenden Implementierung des Hierarchical-Radiosity-Algorithmus. Das *Final-Gathering* [Rei92] wird für die Erzeugung von „Schnappschüssen“ hoher visueller Qualität eingesetzt. Es wird gezeigt, wie dabei die visuelle Qualität der Darstellung von gekrümmten Objekten unter Ausnutzung von Objekt-Informationen

verbessert werden kann. Ferner werden einige Beschleunigungs-Techniken untersucht.

Im Kapitel 4 wird schließlich die Implementierung des Wavelet-Radiance-Algorithmus beschrieben. Es wird eine *wrapper-Datenstruktur* vorgestellt, die eine einfache Anbindung eines Wavelet-Radiance-Algorithmus an ein Wavelet-Radiosity-System ermöglicht. Insbesondere lassen sich auch im Radiance-Fall die berechneten Lösungen sehr effizient durch Gouraud-Shading geglättet ausgeben. Diffuse Reflektoren werden gesondert behandelt.

Das letzte Kapitel enthält eine Zusammenfassung und einen Ausblick. Im Anhang sind die Programm-Optionen dokumentiert.

Kapitel 2

Theoretischer Hintergrund

In diesem Kapitel wird zunächst das globale Beleuchtungsproblem durch ein physikalisch basiertes Beleuchtungsmodell formalisiert. Während der Wavelet-Radiance-Algorithmus unmittelbar auf diesem allgemeinen Beleuchtungsmodell basiert, liegt dem Wavelet-Radiosity-Algorithmus eine spezielle Variante dieses Modells zugrunde. Die gesuchte Lichtverteilung in der Szene ist in beiden Fällen implizit durch eine Integralgleichung (*Radiance-* bzw. *diffuse Radiance-Gleichung*) definiert. Anschließend werden wichtige mathematische Werkzeuge eingeführt, die im folgenden für die effiziente Berechnung einer approximativen Lösung einer derartigen Transportgleichung eingesetzt werden: die *Galerkin-Methode* zur Diskretisierung der kontinuierlichen Transportgleichung, das *Gauss-Seidel-Verfahren* zum iterativen Lösen des resultierenden Linearen Gleichungssystems und *Wavelet-Basen* im Rahmen der Diskretisierung zur deutlichen Reduktion des Zeit- und Speicheraufwands bei einer vorgegebenen Fehlertoleranz. Zuletzt steht die algorithmische Umsetzung des verfolgten Lösungsansatzes im Vordergrund. Das prinzipielle Vorgehen wird durch die abstrakte Beschreibung eines Wavelet-basierten Basis-Algorithmus vorgestellt. Die Beschreibung ist sowohl für den Radiance-Fall als auch für den Radiosity-Fall gültig.

2.1 Physikalisch basiertes Beleuchtungsmodell

Das globale Beleuchtungsproblem [CW93] [SP94] ist ein Simulationsproblem. Es gilt den natürlichen „Transport“ von Licht in einer synthetischen dreidimensionalen Szene zu simulieren. Der umgangssprachliche Begriff „Licht“ korrespondiert physikalisch zu elektromagnetischer Strahlung innerhalb des sichtbaren Wellenlängenbereichs¹. Die physikalischen Phänomene betreffend Ausbreitung und Interaktion mit Materie (z. B. Reflexion an einer Oberfläche) von Licht sind sehr komplex. [Gla95] beinhaltet eine ausführliche Einführung in die entsprechenden physikalischen Hintergründe. In diesem Abschnitt wird ein Beleuchtungsmodell vorgestellt, das sich vor allem durch seinen *globalen Charakter* und die Modellierung von *allgemeinen Reflexionseigenschaften* auszeichnet. Von anderen physikalischen Phänomenen, die nicht unmittelbar mit diesen beiden Hauptaspekten zusammenhängen, wird weitestgehend abstrahiert. Dadurch bleibt das Modell handhabbar, und die Berechnungen bleiben praktikabel. Zunächst werden die wesentlichen Abstraktionen skizziert. Im Anschluß daran wird die geometrische Größe *Raumwinkel* eingeführt, der in der Radiometrie eine große Bedeutung zukommt. Radiometrie bezeichnet dabei die Wissenschaft, Energie-Übertragung durch Strahlung zu messen. Danach wird mit *Radiance* die zentrale radiometrische Größe definiert. Die Reflexionseigenschaften von Oberflächen werden durch die *BRDF* formalisiert. Unter Verwendung dieser Voraussetzungen wird schließlich das dieser Arbeit zugrunde liegende globale Beleuchtungsmodell in Form einer Integralgleichung vorgestellt, die den Transport von Licht in der Szene modelliert.

¹ Sichtbarer Wellenlängenbereich: 380nm – 770nm

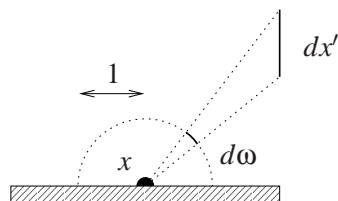
2.1.1 Abstraktionen

Licht wird aus dem Blickwinkel der geometrischen Optik betrachtet. Die geometrische Optik genügt, um deutlich wahrnehmbare makroskopische Lichteffekte wie z. B. Schatten und Reflexion zu erklären. Wellenoptische Lichteffekte wie z. B. Interferenz oder quantenmechanische Lichteffekte werden ignoriert. Das komplette Spektrum wird – wie in der Computergraphik üblich – durch drei ausgezeichnete Wellenlängen² approximiert. Ein Energie-Austausch zwischen verschiedenen Wellenlängen wird nicht berücksichtigt. In diesem Fall können die verschiedenen Wellenlängen unabhängig voneinander betrachtet werden. In der Implementierung werden diskrete Werte von wellenlängenabhängigen Größen³ durch 3-Tupel repräsentiert. Für die weiteren theoretischen Überlegungen wird der monochrome Fall vorausgesetzt. Zwischen den geometrischen Objekten befinde sich kein partizipierendes Medium, das den Lichtfluß beeinflusst. Es genügt, den Lichtfluß an den Oberflächen der geometrischen Objekte zu betrachten. Ferner befinden sich keine transparenten Objekte in der Szene. Die Interaktion von Licht mit Oberflächen beschränkt sich somit auf Reflexion und Absorption. Transmission wird nicht berücksichtigt. Schließlich ist ausschließlich der *steady state* von Interesse. Dieser Zustand stellt sich in realistisch dimensionierten Szenen wegen der hohen Lichtgeschwindigkeit sehr schnell ein. Die Zeitabhängigkeit der radiometrischen Größen wird ignoriert.

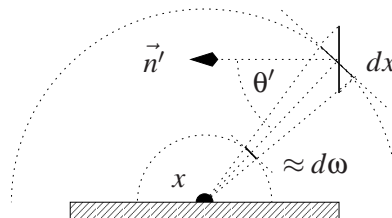
2.1.2 Raumwinkel

Der von einem Objekt bzgl. eines Punktes x aufgespannte *Raumwinkel* ω ist definiert als Flächeninhalt der Normalprojektion des Objekts auf die Einheitskugel mit Mittelpunkt x . Raumwinkel werden in *Steradian* (sr) angegeben. Diese Definition ist eine natürliche dreidimensionale Erweiterung des „normalen“ zweidimensionalen Winkels⁴ (vgl. Abbildung 2.1(a)). Der von einer beliebig im Raum orientierten Differential-Fläche⁵ dx' bzgl. x aufgespannte Differential-Raumwinkel $d\omega$ läßt sich wie folgt gut approximieren (vgl. Abbildung 2.1(b)):

$$d\omega \approx \frac{dx' \cos \theta'}{\|x - x'\|^2} \quad (2.1)$$



(a) exakter Raumwinkel



(b) approximativer Raumwinkel

Abbildung 2.1 Raumwinkel von dx' bzgl. x (flatland)

²Rot, Grün und Blau

³Die radiometrischen Größen und die BRDF sind wellenlängenabhängig.

⁴In der Ebene erfolgt die Projektion auf den Einheitskreis, und der aufgespannte Winkel ist durch die resultierende Bogenlänge definiert.

⁵Im folgenden wird ein Oberflächenpunkt x mit einer Differential-Fläche dx um x identifiziert, deren Normale mit der Oberflächennormale an der Stelle x übereinstimmt – und umgekehrt. Des Weiteren wird mit dx auch der Flächeninhalt der Differential-Fläche dx bezeichnet. Richtungen $\vec{\omega}$ und Differential-Raumwinkel $d\omega$ um $\vec{\omega}$ werden ebenfalls miteinander identifiziert.

Der Zähler kann geometrisch als Orthogonalprojektion von dx' auf die Tangentialebene zu der Kugel mit Mittelpunkt x durch x' interpretiert werden. Bei dieser Formel handelt es sich „nur“ um eine Approximation, da die Projektion nicht direkt auf die Kugel erfolgt. Der Nenner steht für die weitere Projektion auf eine Tangentialebene zur Einheitskugel. Die Beziehung 2.1 wird später benötigt, um das Integral der Radiance-Gleichung von einem Integral über alle Richtungen in ein Integral über alle Oberflächen umzuformen.

2.1.3 Radiometrische Größen

Die zentrale radiometrische Größe ist *Radiance*. Formal ist Radiance in Abhängigkeit eines Oberflächenpunktes x und einer Richtung $\vec{\omega}$ definiert als Strahlungsfluß pro Einheitsraumwinkel und pro orthogonal zur Flußrichtung projizierter Einheitsfläche:

$$\text{Radiance } L(x, \vec{\omega}) \equiv \frac{d^2\Phi(x, \vec{\omega})}{d\omega dx \cos \theta} \quad \left[\frac{W}{sr m^2} \right] \quad (2.2)$$

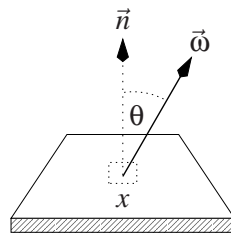


Abbildung 2.2 Definition Radiance: geometrische Situation

Die geometrische Situation ist in Abbildung 2.2 dargestellt. Radiance ist eine vierdimensionale Funktion: Oberflächenpunkte können eindeutig durch zwei skalare *Positionsparameter* identifiziert werden und Richtungen durch zwei skalare *Richtungsparameter*. Da keine transparenten Objekte berücksichtigt werden, sind nur die Richtungen in die äußere Hemisphäre Ω von Interesse. Im folgenden ist mit Hemisphäre stets die äußere Hemisphäre über einem Oberflächenpunkt gemeint. Informell gibt Radiance an, wieviel Licht von einem Oberflächenpunkt in eine Richtung ausstrahlt bzw. wieviel Licht auf einen Oberflächenpunkt aus einer Richtung einstrahlt. Die Begriffe *Outgoing Radiance* (L_o) und *Incoming Radiance* (L_i) werden zur Unterscheidung von ausstrahlender und einstrahlender Radiance verwendet [Laf96]. Radiance hat die wichtige Eigenschaft, invariant entlang der Flußrichtung zu sein:

$$\forall \text{ gegenseitig zugewandten und sichtbaren } x, x' : L_i(x, x' - x) = L_o(x', x - x') \quad (2.3)$$

Diese Eigenschaft folgt aus der allgemeineren Energie-Erhaltungseigenschaft unter der Voraussetzung, daß der Lichtfluß durch kein partizipierendes Medium zwischen den Oberflächen beeinflusst wird. Da zudem das menschliche visuelle System gerade Radiance-sensitiv ist, wird der visuelle Eindruck einer Szene vollständig durch die von jedem (sichtbaren) Oberflächenpunkt in Richtung Augpunkt strahlende Outgoing Radiance beschrieben. Daher ist Outgoing Radiance die geeignete radiometrische Größe, die es von globalen Beleuchtungsalgorithmen zu berechnen gilt. Alle weiteren für diese Arbeit relevanten radiometrischen Größen lassen sich von Radiance ableiten (vgl. Tabelle 2.1).

Bezeichnung	Definition	Einheit
Radiosity $B(x)$	$\int_{\Omega} L_o(x, \vec{\omega}) \cos \theta d\omega$	
Irradiance $E(x)$	$\int_{\Omega} L_i(x, \vec{\omega}) \cos \theta d\omega$	$\frac{W}{m^2}$
Differential-Irradiance $dE(x, d\omega)$	$L_i(x, \vec{\omega}) \cos \theta d\omega$	

Tabelle 2.1 Von Radiance abgeleitete radiometrische Größen

2.1.4 BRDF

Die Reflexionseigenschaften der Oberflächen werden durch die *Bidirectional-Reflectance-Distribution-Function (BRDF)* quantifiziert. Formal ist die BRDF in Abhängigkeit eines Oberflächenpunktes x , einer Reflexionsrichtung $\vec{\omega}_r$ und einer Einstrahlrichtung $\vec{\omega}_i$ definiert als Verhältnis von in die Reflexionsrichtung reflektierter Differential-Radiance zu aus der Einstrahlrichtung einfallender Differential-Irradiance:

$$\text{BRDF } f_r(x, \vec{\omega}_r, \vec{\omega}_i) \equiv \frac{dL_r(x, \vec{\omega}_r)}{L_i(x, \vec{\omega}_i) \cos \theta_i d\omega_i} \quad \left[\frac{1}{sr} \right] \quad (2.4)$$

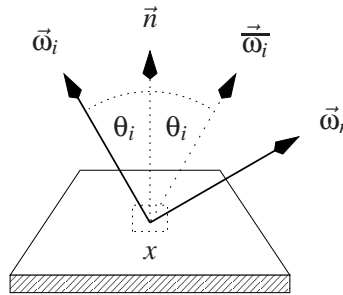


Abbildung 2.3 Definition BRDF: geometrische Situation

Abbildung 2.3 zeigt die geometrische Situation. Informell gibt die BRDF für einen Oberflächenpunkt an, welcher Anteil des aus einer Richtung einstrahlenden Lichts in eine zweite Richtung reflektiert wird.

Die BRDF hat zwei wichtige Eigenschaften, die unmittelbar aus entsprechenden thermodynamischen Prinzipien folgen (vgl. Tabelle 2.2): Energie-Erhaltung und Helmholtz-Reziprozität. Üblicherweise wird die BRDF durch eine effizient zu berechnende Funktion approximiert. Ein derartiges Reflexionsmodell gilt als *physikalisch plausibel*, wenn es die beiden o. g. Eigenschaften erfüllt. Ein physikalisch basiertes Beleuchtungsmodell setzt ein physikalisch plausibles Reflexionsmodell voraus. Das in der Implementierung des Wavelet-Radiance-Algorithmus verwendete physikalisch plausible Reflexionsmodell [LW94] wird im Abschnitt 4.1.5.1 vorgestellt.

Energie-Erhaltung	Helmholtz-Reziprozität
$\int_{\Omega} f_r(x, \vec{\omega}_r, \vec{\omega}_i) \cos \theta_r d\omega_r \leq 1 \quad \forall x, \vec{\omega}_i$ Es wird nicht mehr Energie in die Hemisphäre reflektiert, als Energie einfällt. Die nicht-reflektierte Energie wird absorbiert. Diese Eigenschaft ist eine notwendige Bedingung für die Konvergenz von iterativen globalen Beleuchtungsalgorithmen.	$f_r(x, \vec{\omega}_r, \vec{\omega}_i) = f_r(x, \vec{\omega}_i, \vec{\omega}_r) \quad \forall x, \vec{\omega}_r, \vec{\omega}_i$ Die BRDF ist invariant gegenüber einer Vertauschung von Reflexions- und Einstrahlrichtung.

Tabelle 2.2 Eigenschaften der BRDF

Im Radiosity-Fall besteht die Szene ausschließlich aus diffusen Reflektoren. Diffuse Reflektoren reflektieren in alle möglichen Richtungen stets dieselbe Radiance, d. h. die BRDF ist unabhängig von der Reflexionsrichtung. Aus der Helmholtz-Reziprozität folgt weiter, daß die BRDF im Radiosity-Fall für jeden Oberflächenpunkt sogar konstant ist.

2.1.5 Radiance-Gleichung

Mit Hilfe der eingeführten Definitionen und Folgerungen läßt sich nun die zentrale *Radiance-Gleichung* herleiten, durch die die gesuchte Outgoing Radiance implizit definiert wird:

$$L_o(x, \vec{\omega}) = L_e(x, \vec{\omega}) + \int_{\vec{\omega}_i \in \Omega} f_r(x, \vec{\omega}, \vec{\omega}_i) L_o(x', -\vec{\omega}_i) \cos \theta_i d\omega_i$$

Die Outgoing Radiance setzt sich aus einem emittierenden und einem reflektierenden Anteil zusammen. Bei der Ermittlung des reflektierenden Anteils werden *alle* möglichen Einstrahlrichtungen berücksichtigt. Für eine feste Einstrahlrichtung läßt sich die reflektierende Radiance direkt aus der Definition der BRDF 2.4 ableiten. Unter Ausnutzung der Invarianz-Eigenschaft 2.3 wird die Incoming Radiance auf Punkt x durch die korrespondierende Outgoing Radiance von Punkt x' ersetzt, wobei x' den von x aus am nächsten in Richtung $\vec{\omega}_i$ gelegenen Oberflächenpunkt bezeichnet. Im folgenden ist mit Radiance (L) grundsätzlich Outgoing Radiance (L_o) gemeint. Die Integration über die komplette Hemisphäre und die Rekursion machen den globalen Charakter der Radiance-Gleichung aus. Die Verwendung der BRDF läßt die Modellierung von allgemeinen Reflexionseigenschaften zu. Unter Ausnutzung der Beziehung 2.1 kann das Integral über alle Richtungen in ein Integral über alle Oberflächen umgeformt werden. Im folgenden dient diese Form der Radiance-Gleichung als Grundlage zur Ermittlung einer Lösung:

$$L(x, \vec{\omega}) = L_e(x, \vec{\omega}) + \int_{x' \in S} f_r(x, \vec{\omega}, x' - x) g(x, x') L(x', x - x') dx'$$

S bezeichnet die Menge aller Oberflächenpunkte. Die geometrischen Größen werden dabei zu einem *geometrischen Term* g zusammengefaßt:

$$\text{geometrischer Term } g(x, x') \equiv v(x, x') \frac{\cos \theta \cos \theta'}{\|x - x'\|^2} \quad (2.5)$$

Die bool'sche *Sichtbarkeits-Funktion* v gibt an, ob die beiden Punkte gegenseitig sichtbar sind. Dies ist genau dann der Fall, wenn sich zwischen den beiden Punkten kein blockierendes Objekt befindet. Abbildung 2.4 zeigt die geometrische Situation. Der Strahlungsfluß zwischen zwei Differential-Flächen ist proportional zu ihrem geometrischen Term⁶. Der geometrische Term ist symmetrisch. Er nimmt quadratisch mit der Entfernung der beiden Differential-Flächen ab. Je orthogonaler die beiden Differential-Flächen zu ihrer Verbindungslinie liegen, desto größer ist der geometrische Term.

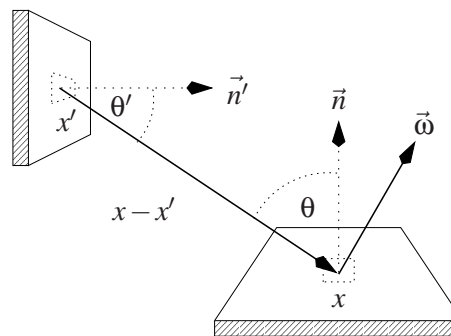


Abbildung 2.4 Radiance-Gleichung: geometrische Situation

⁶ Aus 2.1, 2.2 und 2.5 folgt: $d^2\Phi(x, x' - x) = L(x, x' - x) g(x, x') dx dx'$

Im Radiosity-Fall ist die gesuchte Radiance richtungsunabhängig, da sowohl emittierende Radiance als auch reflektierende Radiance richtungsunabhängig sind. Die Radiance-Gleichung vereinfacht sich zu der folgenden *diffusen Radiance-Gleichung*:

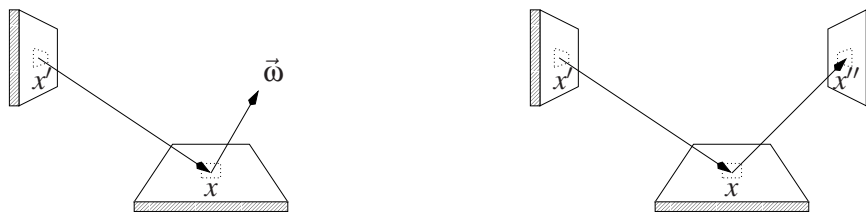
$$L(x) = L_e(x) + f_r(x) \int_{x' \in S} g(x, x') L(x') dx'$$

Die Struktur der Gleichung bleibt erhalten. Allerdings reduziert sich die Dimensionalität der gesuchten Funktion von vier auf zwei. Dieser Fall wird als „Radiosity-Fall“ bezeichnet, da die Radiance in diesem Fall proportional zu der per Definition richtungsunabhängigen radiometrischen Größe Radiosity ist. Folglich genügt es, die zweidimensionale Radiosity anstelle der im allgemeinen vierdimensionalen Radiance zu berechnen.

Die Radiance-Gleichung ist eng mit der klassischen Rendering-Gleichung [Kaj86] verwandt. Beide Gleichungen haben dieselbe Struktur. Im Gegensatz zur Radiance-Gleichung basiert die Rendering-Gleichung nicht auf der radiometrischen Größe Radiance, sondern auf der – ebenfalls vierdimensionalen – *two-point-intensity* $L^*(x, x')$. Die beiden Größen stehen in der folgenden Beziehung zueinander:

$$L^*(x, x') = g(x, x') L(x, x' - x)$$

Den beiden Gleichungen liegen verschiedene Transportmodelle zugrunde (vgl. Abb. 2.5). Die Radiance-Gleichung korrespondiert zu Transporten von einem Punkt über einen Punkt *in eine Richtung*. Demgegenüber korrespondiert die Rendering-Gleichung zu Transporten von einem Punkt über einen Punkt *zu einem Punkt*. Der als Grundlage dieser Arbeit ausgewählte Wavelet-Radiance-Algorithmus von Christensen basiert – wie auch der nicht-Wavelet-basierte Algorithmus von Immel [ICG86] – auf der Radiance-Gleichung. Andere Wavelet-Radiance-Algorithmen basieren auf der Rendering-Gleichung [AH93] [Sch94b]. Die Auswahl der Transportgleichung hat einen großen Einfluß auf die Gestalt und die Eigenschaften der resultierenden Wavelet-basierten Algorithmen (vgl. Kapitel 5).



(a) gerichteter Transport (Radiance-Gleichung)

(b) 3-Punkt-Transport (Rendering-Gleichung)

Abbildung 2.5 Transportmodelle im Radiance-Fall

2.2 Mathematische Werkzeuge

Im vorigen Abschnitt wurde das globale Beleuchtungsproblem durch eine Fredholm-Integralgleichung der zweiten Art formalisiert. Die Sprache der *linearen Operatoren* hat sich als ein sehr nützliches Werkzeug für die Beschreibung von Integralgleichungen erwiesen, da die resultierenden Ausdrücke übersichtlicher sind und die algebraischen Eigenschaften betonen. Die Radiance-Gleichung läßt sich wie folgt in der Sprache der linearen Operatoren ausdrücken⁷:

$$L = L_e + \mathcal{T}L \quad \text{bzw.} \quad \mathcal{M}L = L_e \quad \text{mit} \quad \mathcal{M} \equiv \mathcal{I} - \mathcal{T}$$

\mathcal{T} bezeichnet dabei den linearen *Transport-* bzw. *Kerneloperator*, der wie folgt definiert ist:

$$\begin{aligned} (\mathcal{T}L)(x, \vec{\omega}) &\equiv \int_{x' \in S} k(x, \vec{\omega}, x') L(x', x - x') dx' \\ &\quad \text{mit Kernel } k(x, \vec{\omega}, x') \equiv f_r(x, \vec{\omega}, x' - x) g(x, x') \quad (\text{Radiance-Fall}) \\ (\mathcal{T}L)(x) &\equiv \int_{x' \in S} k(x, x') L(x') dx' \\ &\quad \text{mit Kernel } k(x, x') \equiv f_r(x) g(x, x') \quad (\text{Radiosity-Fall}) \end{aligned}$$

Die Anwendung des Transportoperators kann physikalisch als *ein* globaler Lichttransport interpretiert werden: Der Transportoperator ordnet einer gegebenen Lichtverteilung diejenige Lichtverteilung zu, die sich nach der einmaligen Ausbreitung von Licht in die Umgebung und der anschließenden Reflexion an der Umgebung einstellt.

Das Lösen der Radiance-Gleichung erfordert formal die Invertierung des linearen Operators \mathcal{M} . Da bei der Interaktion von Licht mit einer Oberfläche stets Energie absorbiert wird, gilt $\|\mathcal{T}\| < 1$. In diesem Fall existiert die Inverse von \mathcal{M} und $\mathcal{M}^{-1}L_e$ ist die exakte Lösung der Radiance-Gleichung. Ferner gilt:

$$\mathcal{M}^{-1} = \sum_{i=0}^{\infty} \mathcal{T}^i$$

Diese *Neumann-Reihe* ist das Operator-Analogon zu der geometrischen Reihe⁸. Im Kontext des globalen Beleuchtungsproblems besitzt die Neumann-Reihe eine anschauliche physikalische Interpretation. Sie formalisiert die Lösungs idee, alle möglichen Lichttransporte in die Umgebung ausgehend von einer den Lichtquellen zugeordneten Ausgangs-Lichtverteilung zu simulieren. Die Radiance-Gleichung läßt sich in der Regel nicht analytisch exakt lösen, da u. a. die geometrischen Verhältnisse in nicht-trivialen Szenen zu komplex sind. Statt dessen begnügt man sich mit der Berechnung einer *approximativen* Lösung \tilde{L} durch eine *numerische Methode*. Die formale Problembeschreibung wird wie folgt angepaßt:

$$\tilde{L} \approx L_e + \mathcal{T}\tilde{L} \quad \text{bzw.} \quad \mathcal{M}\tilde{L} \approx L_e \quad \text{bzw.} \quad R \approx 0 \quad \text{mit} \quad \text{Residuum } R \equiv L_e - \mathcal{M}\tilde{L}$$

Zunächst wird die mathematische Struktur skizziert, die von der eingesetzten Galerkin-Methode vorausgesetzt wird. Im Anschluß daran wird diese numerische Methode ausführlich beschrieben. Die Galerkin-Methode überführt die *kontinuierliche* Radiance-Gleichung durch die Verwendung einer endlichen Basis in eine *diskrete* Radiance-Gleichung. Die ersten beiden Abschnitte basieren auf [Arv93]. Danach wird das Gauss-Seidel-Verfahren zum iterativen Lösen der diskreten Radiance-Gleichung vorgestellt. Schließlich wird der Einsatz von Wavelet-Basen bei der Diskretisierung der Radiance-Gleichung nach der Galerkin-Methode beschrieben.

⁷ Die Parameter werden üblicherweise weggelassen.

⁸ Geometrische Reihe: $(1-x)^{-1} = \sum_{i=0}^{\infty} x^i$ für $|x| < 1$

2.2.1 Hilbert-Raum $\mathcal{L}^2(D)$

Als Ausgangs-Vektorraum für die mathematische Handhabung von Radiance-Verteilungen diene der unendlich-dimensionale Vektorraum $\mathcal{L}^2(D)$ bestehend aus allen *quadratisch integrierbaren* Funktionen über dem Definitionsbereich D mit:

$$D \equiv \begin{cases} S \times \Omega, & \text{Radiance-Fall} \\ S, & \text{Radiosity-Fall} \end{cases}$$

Die beiden Vektorraum-Operationen seien auf natürliche Weise (d. h. punktweise) definiert. Eine Funktion heißt quadratisch integrierbar, wenn ihr durch die folgende Abbildung ein endlicher Wert zugeordnet wird:

$$F \mapsto \left[\int_D |F(x)|^2 dx \right]^{1/2} \quad (2.6)$$

Da die Elemente von $\mathcal{L}^2(D)$ Funktionen sind, spricht man auch von einem Funktionenraum. Das sukzessive Hinzufügen von weiteren Eigenschaften führt zu einer Hierarchie von Vektorraum-Klassen mit zunehmender mathematischer Struktur: normierte Vektorräume, Banach-Räume und Hilbert-Räume. Die Galerkin-Methode zum approximativen Lösen von Integralgleichungen setzt die starke Hilbert-Raum-Struktur voraus. Zuordnung 2.6 ist eine Norm – die sog. 2-Norm $\|\cdot\|_2$ – auf $\mathcal{L}^2(D)$. Darüber hinaus ist der mit der 2-Norm ausgestattete Vektorraum $\mathcal{L}^2(D)$ ein Banach-Raum, da auch die Vollständigkeits-Eigenschaft erfüllt ist. Der mit der 2-Norm ausgestattete Vektorraum $\mathcal{L}^2(D)$ ist sogar ein Hilbert-Raum, da zudem die 2-Norm von dem wie folgt definierten (Standard-)Skalarprodukt für Funktionenräume induziert wird:

$$\langle F, G \rangle \equiv \int_D F(x)G(x) dx$$

Hilbert-Räume haben einen „geometrischen Beigeschmack“, da sich mit Hilfe des Skalarprodukts *Orthogonalität* und *Orthogonalprojektion* definieren lassen. Zwei Funktionen $F, G \in \mathcal{L}^2(D)$ sind genau dann *orthogonal* zueinander ($F \perp G$), wenn ihr Skalarprodukt Null ergibt. Die Orthogonalprojektion auf eine Basisfunktion $B \in \mathcal{L}^2(D)$ ist wie folgt definiert:

$$F \mapsto \langle F, B^* \rangle B$$

B^* bezeichnet die zu B *duale* Basisfunktion. Für die weiteren theoretischen Überlegungen werden stets Orthonormalbasen vorausgesetzt. Da Orthonormalbasen selbst-dual sind, kann B^* in der o. g. Definition durch B ersetzt werden.

Für das Verständnis der Galerkin-Methode ist von zentraler Bedeutung, daß die Elemente eines Hilbert-Raums durch die Projektion in einen geeigneten (endlich-dimensionalen) Unterraum approximiert werden können.

2.2.2 Galerkin-Methode

Die Galerkin-Methode gehört zu der Klasse der *Finite-Basis-Methoden*. Das Grundkonzept von Finite-Basis-Methoden besteht darin, die Suche nach einer geeigneten approximativen Lösung der Integralgleichung auf einen vorgegebenen n -dimensionalen Unterraum $X_n \subset \mathcal{L}^2(D)$ zu beschränken. Finite-Basis-Methoden führen schließlich zu einem *Linearen Gleichungssystem* (LGS), durch das die n Koeffizienten $\{L_1, \dots, L_n\}$ von \tilde{L} bzgl. einer vorgegebenen Basis von X_n implizit definiert sind. Alle Finite-Basis-Methoden haben wenigstens die folgenden beiden Freiheitsgrade gemeinsam, die die Güte der Approximation und die Effizienz der Methode entscheidend beeinflussen:

Auswahl des Unterraums X_n Die exakte Lösung $\mathcal{M}^{-1}L_e \in \mathcal{L}^2(D)$ kann nur durch Funktionen in X_n approximiert werden.

Auswahl einer Basis für X_n Die Komplexität der Methode kann z. B. durch den Einsatz einer Wavelet-Basis asymptotisch reduziert werden.

Für die folgenden Betrachtungen seien ein geeigneter Unterraum X_n und eine geeignete Basis $\{B_1, \dots, B_n\}$ für X_n fixiert. Verschiedene Kriterien zur formalen Definition von „ $R \approx 0$ “ – und damit zur Auswahl von $\tilde{L} \in X_n$ – führen zu verschiedenen Finite-Basis-Methoden (vgl. Tabelle 2.3). Die Kriterien der Least-Squares-Methode und der Galerkin-Methode unterscheiden sich nur im involvierten Unterraum: Bild von X_n unter \mathcal{M} bzw. X_n selbst. Das Bild von X_n unter \mathcal{M} ist ein Unterraum $Y_m \subset \mathcal{L}^2(D)$ der Dimension m , $m \leq n$.

Finite-Basis-Methoden	Kriterien zur Auswahl der Approximation
Collocation	Sei $\{x_1, \dots, x_n\} \subset D$ $R(x_i) = 0 \quad \forall 1 \leq i \leq n$
Least-Squares	$R \perp Y_m$ mit $Y_m \equiv \text{Im } \mathcal{M} _{X_n}$
Galerkin	$R \perp X_n$

Tabelle 2.3 Finite-Basis-Methoden und zugrunde liegende Approximations-Kriterien

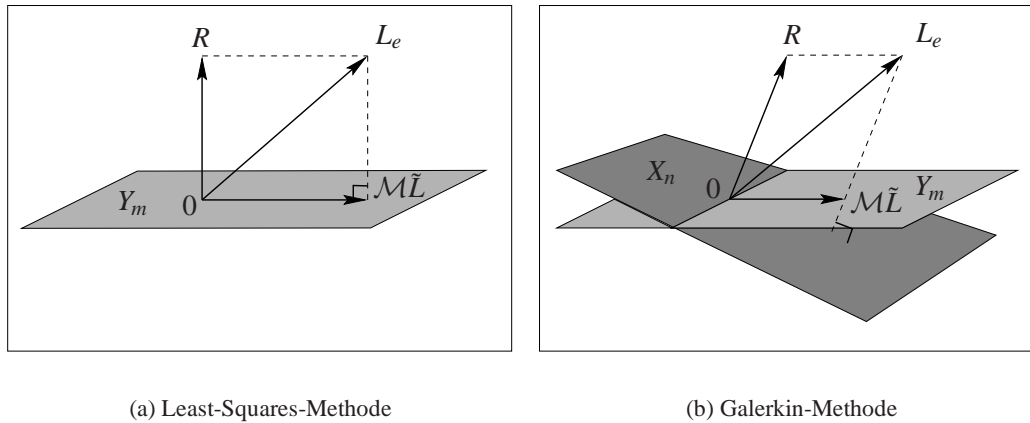


Abbildung 2.6 Geometrische Darstellung der Projektionsmethoden

Die Least-Squares-Methode ist eine *Projektionsmethode*, da L_e orthogonal auf Y_m projiziert wird (vgl. Abbildung 2.6(a)). Unter allen Funktionen in X_n wird eine Funktion \tilde{L} mit in bezug auf die 2-Norm minimalem korrespondierendem Residuum ausgewählt. In diesem Sinn selektiert die Least-Squares-Methode eine *optimale*⁹ Approximation.

Die Galerkin-Methode zählt ebenfalls zu den Projektionsmethoden. Sie projiziert L_e orthogonal auf X_n und somit i. a. schief auf Y_m (vgl. Abbildung 2.6(b)). Die Approximation ist daher im Least-Squares-Sinn i. a. *nicht* optimal. Falls jedoch X_n und Y_m „ähnlich“ sind¹⁰, so ist die

⁹ Wünschenswert wäre die Minimierung des tatsächlichen Fehlers $L - \tilde{L}$. Da L jedoch nicht bekannt ist, begnügt man sich mit der Minimierung des „abgebildeten Fehlers“ $\mathcal{M}(L - \tilde{L}) = R$. Das Bild von L ist bekannt: $\mathcal{M}L = L_e$

¹⁰ d. h. \mathcal{M} bildet X_n „größtenteils“ auf sich selbst ab

Approximation wenigstens nahezu optimal. Der entscheidende praktische Vorteil gegenüber der Least-Squares-Methode besteht darin, daß die Matrix-Einträge des resultierenden LGSs leichter zu berechnen sind. Das zu der Galerkin-Methode korrespondierende LGS (die *diskrete Radiance-Gleichung*) läßt sich aus dem Galerkin-Kriterium zur Auswahl der Approximation herleiten (vgl. Tabelle 2.4):

$$L = L_e + TL$$

	$R \perp X_n$	
\Leftrightarrow	$R \perp B_r$	$\forall 1 \leq r \leq n$
Def. Orthogonalität		
\Leftrightarrow	$\langle L_e - \mathcal{M}\tilde{L}, B_r \rangle = 0$	$\forall 1 \leq r \leq n$
Bilinearität des Skalarprodukts		
\Leftrightarrow	$\langle \mathcal{M}(\sum_{s=1}^n L_s B_s), B_r \rangle = \langle L_e, B_r \rangle$	$\forall 1 \leq r \leq n$
Lin. von \mathcal{M} , Bilin. und Sym. des SPs		
\Leftrightarrow	$\sum_{s=1}^n L_s \langle B_r, \mathcal{M}B_s \rangle = \langle L_e, B_r \rangle$	$\forall 1 \leq r \leq n$
Bilin. des SPs, Orthonorm. der Basis		
\Leftrightarrow	$L_r = \langle L_e, B_r \rangle + \sum_{s=1}^n \langle B_r, \mathcal{T}B_s \rangle L_s$	$\forall 1 \leq r \leq n$

Tabelle 2.4 Herleitung des zu dem Galerkin-Kriterium korrespondierenden LGSs

L bzw. L_e bezeichnet den n -dimensionalen Spaltenvektor bestehend aus den Koeffizienten von \tilde{L} bzw. der Orthogonalprojektion von L_e auf X_n bzgl. der Basis. Die Einträge der quadratischen *Transportmatrix* T werden *Transportkoeffizienten* genannt:

$$\begin{aligned}
 T_{r \leftarrow s} &\equiv \langle B_r, \mathcal{T}B_s \rangle \\
 &= \begin{cases} \int_{x \in S} \int_{\vec{\omega} \in \Omega} B_r(x, \vec{\omega}) \int_{x' \in S} f_r(x, \vec{\omega}, x') g(x, x') B_s(x', x - x') dx' d\vec{\omega} dx, & \text{Radiance-Fall} \\ \int_{x \in S} B_r(x) f_r(x) \int_{x' \in S} g(x, x') B_s(x') dx' dx, & \text{Radiosity-Fall} \end{cases}
 \end{aligned}$$

Ein Transportkoeffizient gibt den Einfluß einer *Sender*-Basisfunktion auf eine *Empfänger*-Basisfunktion an. Die ungewöhnliche Notation der Indizes soll an diese Interpretation erinnern. Die Integration kann auf den Träger der Empfänger-Basisfunktion und den Träger¹¹ der Sender-Basisfunktion beschränkt werden. Üblicherweise werden Basisfunktionen mit einem sehr kompakten Träger eingesetzt, um die Berechnung der Transportkoeffizienten zu vereinfachen. In diesem Fall spricht man auch von Finite-Elemente-Methoden.

Die Kompaktheit der Träger führt außerdem dazu, daß die Transportmatrix im Radiance-Fall dünnbesetzt ist. Ein Transportkoeffizient ist höchstens dann verschieden von Null, wenn der Positionsträger der Empfänger-Basisfunktion $supp_p(B_r)$ zumindest teilweise innerhalb des Richtungs-trägers der Sender-Basisfunktion $supp_d(B_s)$ liegt (vgl. Abbildung 2.7). Christensen hat nachgewiesen, daß von den n^2 Transportkoeffizienten nur $O(n^{1.5})$ von Null verschieden sein können [Chr95].

Finite-Basis-Methoden wählen eine *approximative* Lösung für eine Integralgleichung aus, die durch eine *driving-function* L_e und einen Kernel-Operator \mathcal{T} definiert ist. Bemerkenswerterweise wählt die Galerkin-Methode dabei gerade die *exakte* Lösung der „in X_n projizierten Integralgleichung“ aus, die durch die projizierte driving-function $\mathcal{P}_n L_e$ und den projizierten Kernel-Operator $\mathcal{P}_n \mathcal{T} \mathcal{P}_n$ definiert ist [Arv93]. Dabei bezeichnet $\mathcal{P}_n : \mathcal{L}^2(D) \rightarrow X_n$ den linearen Projektionsoperator, der eine Funktion orthogonal auf X_n projiziert.

¹¹ genauer: Positionsträger im Radiance-Fall

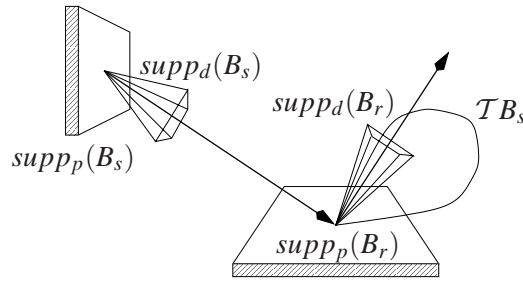


Abbildung 2.7 Transport zwischen Basisfunktionen im Radiance-Fall

2.2.3 Gauss-Seidel-Verfahren

Die approximative Lösung \tilde{L} ist durch die diskrete Radiance-Gleichung implizit definiert. Die explizite Berechnung von L erfolgt durch das Lösen dieses Linearen Gleichungssystems. Die Berechnung der *exakten* Lösung des Gleichungssystems durch ein direktes Verfahren (z. B. Gaußsches Eliminationsverfahren¹²) ist in dem konkreten Anwendungsfall zu aufwendig. Darüber hinaus ist die Berechnung der exakten Lösung auch nicht notwendig, da selbst die exakte Lösung der diskreten Radiance-Gleichung nur eine approximative Lösung der ursprünglichen kontinuierlichen Radiance-Gleichung ist. Statt dessen wird eine *approximative* Lösung des Linearen Gleichungssystems durch ein iteratives Verfahren (*Relaxations-Verfahren*) berechnet. Der Zeitaufwand reduziert sich auf $O(sn^2)$, wobei s der Anzahl der Iterationsschritte entspricht. Durch den Einsatz einer Wavelet-Basis bei der Diskretisierung lassen sich Zeit- und Speicheraufwand weiter reduzieren.

Das *Jacobi-Verfahren* korrespondiert zu der iterativen Anwendung der ersten Terme der *diskreten* Neumann-Reihe auf den Startvektor L_e . Nach i Iterationen sind gerade alle Lichttransporte mit bis zu i aufeinanderfolgenden Reflexionen simuliert:

$$\begin{aligned} L^{(0)} &\equiv L_e \\ L^{(i+1)} &\equiv L_e + TL^{(i)} \end{aligned}$$

Das *Gauss-Seidel-Verfahren* unterscheidet sich von dem Jacobi-Verfahren durch die Berücksichtigung von Zwischenergebnissen während eines Iterationsschrittes:

$$\begin{aligned} L^{(0)} &\equiv L_e \\ L_r^{(i+1)} &\equiv L_{e,r} + \sum_{s=1}^{r-1} T_{r \leftarrow s} L_s^{(i+1)} + \sum_{s=r+1}^n T_{r \leftarrow s} L_s^{(i)} \quad \forall 1 \leq r \leq n \end{aligned}$$

Das Gauss-Seidel-Verfahren benötigt im Vergleich zum Jacobi-Verfahren nur den halben Speicherplatz für die Darstellung der Radiance-Koeffizienten. Bei der Implementierung des Gauss-Seidel-Verfahrens genügt ein Vektor für die Verwaltung von $L^{(i)}$ und $L^{(i+1)}$, da während eines Iterationsschrittes „alte“ Koeffizienten direkt durch „neue“ ersetzt werden können. Das Jacobi-Verfahren erfordert hingegen zwei Vektoren, da die Berechnung der neuen Koeffizienten ausschließlich auf den alten basiert. Erst nach einem kompletten Iterationsschritt werden die alten Koeffizienten durch die neuen ersetzt. Außerdem konvergiert das Gauss-Seidel-Verfahren wegen der Berücksichtigung von Zwischenergebnissen rascher. Nach i Iterationen sind wenigstens alle Lichttransporte mit bis zu i aufeinanderfolgenden Reflexionen simuliert.

¹² Zeitkomplexität: $O(n^3)$, Speicherkomplexität: $O(n^2)$

2.2.4 Wavelet-Basen

Die Galerkin-Methode wurde für einen beliebigen endlich-dimensionalen Approximationsraum und eine beliebige Basis des gewählten Approximationsraums beschrieben. In diesem Abschnitt wird eine Klasse von *hierarchischen* Basen vorgestellt, die sich hauptsächlich aufgrund der folgenden Schlüsseleigenschaft für die Diskretisierung der Radiance-Gleichung anbieten: Die Projektion einer (zumindest teilweise) „glatten“ Funktion in eine Wavelet-Basis führt zu „vielen kleinen“ *unsignifikanten* Koeffizienten. Insbesondere enthält die resultierende Transportmatrix nur „wenige“ *signifikante* Transportkoeffizienten. Das Ignorieren aller unsignifikanten Transportkoeffizienten führt zu einer dünnbesetzten Transportmatrix, die den Transportoperator im Vergleich zu der vollbesetzten Transportmatrix nur geringfügig ungenauer approximiert. Wavelet-basierte globale Beleuchtungsalgorithmen nutzen gerade diesen Umstand zur deutlichen Reduktion des Zeit- und Speicheraufwands aus.

Die Wavelet-Theorie hat Wurzeln in der Approximations-Theorie [Dau88] und in der Signalverarbeitung [Mal89]. Wavelets haben sich als ein sehr nützliches mathematisches Werkzeug zum effizienten Lösen von praktischen Problemen in sehr unterschiedlichen Anwendungsbereichen etabliert. In dieser Arbeit steht der Einsatz von Wavelet-Basen im Rahmen der Galerkin-Methode zum *effizienten* Lösen der Radiance-Integralgleichung im Vordergrund und – gewissermaßen als Vorstufe – die Wavelet-basierte Approximation von *gegebenen* Funktionen.

Zunächst soll der Einsatz von Wavelet-Basen zum effizienten Lösen des globalen Beleuchtungsproblems motiviert werden. Danach erfolgt eine Einführung in die Wavelet-basierte Funktionsapproximation anhand der einfachsten Wavelet-Basis – der eindimensionalen *Haar-Basis*. Anschließend wird die Einführung auf den mehrdimensionalen Fall erweitert, da sowohl die mehrdimensionale Radiance-Verteilung als auch der mehrdimensionale Kernel in einer Wavelet-Basis approximiert werden sollen. Danach wird der Einsatz der Haar-Basis im Rahmen der Diskretisierung der Radiance-Gleichung erläutert. Der von Christensen und in dieser Arbeit verfolgten Standard-Zerlegung des Transportoperators wird die üblicherweise verwendete Nicht-Standard-Zerlegung gegenübergestellt. Viele der im Zusammenhang mit der Haar-Basis eingeführten Notationen, Konstruktionen und Eigenschaften (z. B. die o. g. Schlüsseleigenschaft) lassen sich auch auf andere Wavelet-Basen übertragen. Die verschiedenen Wavelet-Basen unterscheiden sich z. B. im Approximationsraum oder in der Interpretation von „glatt“. Schließlich wird die Auswahl der Haar-Basis aus der Klasse aller Wavelet-Basen für die Diskretisierung der Radiance-Gleichung diskutiert.

2.2.4.1 Motivation

Die ersten Finite-Elemente-Methoden zum Lösen des globalen Beleuchtungsproblems waren nicht Wavelet-basiert¹³. Die Ausführungen in diesem Abschnitt beziehen sich o. B. d. A. auf den Radiosity-Fall. Die Oberflächen der geometrischen Objekte seien jeweils durch ein Netz von planaren Ausgangs-Patches approximiert. Jeder Ausgangs-Patch wird a priori in eine Menge von *Elementen* unterteilt (*Meshing*) und zu jedem Element wird ein mittlerer diffuser Radiance-Wert verwaltet. Diese Vorgehensweise läuft auf den Einsatz einer kanonischen Basis bestehend aus Box-Funktionen hinaus, deren Träger mit den Elementen übereinstimmen. Der Approximationsraum umfaßt folglich alle über den Elementen konstanten Funktionen.

Diese nicht-hierarchischen Algorithmen sind hinsichtlich der folgenden beiden Aspekte nicht optimal:

Meshing Optimal wäre eine feine Unterteilung in Bereichen mit großen Radiance-Gradienten und eine grobe Unterteilung in den übrigen Bereichen. Im Rahmen eines a priori Meshings

¹³ [ICG86] (Radiance-Fall) bzw. [GTGB84] (Radiosity-Fall)

werden die Ausgangs-Patches jedoch häufig vergleichsweise entweder zu grob oder zu fein unterteilt. Eine zu grobe Unterteilung hat einen negativen Einfluß auf die Genauigkeit und eine zu feine Unterteilung wirkt sich ungünstig auf die Effizienz aus.

Transport zwischen Basisfunktionen Selbst unter der Voraussetzung eines optimalen Meshings geht sehr viel Effizienz dadurch verloren, daß zwischen *allen* Basisfunktionen Licht transportiert wird. Zur Veranschaulichung diene eine synthetische Szene bestehend aus einer Wand und zwei Lichtquellen mit unterschiedlichen Abständen zur Wand. Der Transport von der nah platzierten Lichtquelle zur Wand erfordert eine feine Unterteilung der Wand, um die resultierende stark variierende Lichtverteilung gut approximieren zu können. Folglich wird die Wand bei einem optimalen Meshing fein unterteilt. Der Transport von der entfernt platzierten Lichtquelle zur Wand erfordert hingegen keine feine Unterteilung der Wand, da die resultierende Lichtverteilung kaum variiert. Trotzdem wird auch in diesem Fall Energie zu allen Basisfunktionen der Wand transportiert. Dieser Lichttransport wird „zu genau“ berechnet.

Wavelet-basierte Algorithmen verwalten für jeden Ausgangs-Patch – neben einer *Glätt*efunktion – eine dynamisch wachsende Hierarchie von *Detail*funktionen. Das zu den feinsten Auflösungsstufen der Hierarchien korrespondierende Meshing strebt mit der Zeit adaptiv gegen ein optimales Meshing. Die Transporte zwischen den Ausgangs-Patches finden dabei auf einem hinsichtlich einer vorgegebenen Fehlertoleranz geeigneten *level-of-detail* statt. Die synthetische Szene diene weiter als Beispiel: Von der nah platzierten Lichtquelle wird Energie zu allen Basisfunktionen der Wand-Hierarchie transportiert. Die Transporte entsprechen qualitativ und quantitativ den Transporten zu allen Box-Funktionen im nicht-hierarchischen Fall. Demgegenüber wird von der entfernt platzierten Lichtquelle Energie nur zu der Glättfunktion und zu „wenigen groben“ Detailfunktionen in der Spitze der Wand-Hierarchie transportiert. Die Transporte zu den „vielen feinen“ Detailfunktionen können ignoriert werden, da sie – in diesem Fall aufgrund der Entfernung – unsignifikant sind.

2.2.4.2 Eindimensionale Haar-Basis

In diesem Abschnitt stehe das folgende Problem im Vordergrund: Gegeben seien eine Funktion $f \in \mathcal{L}^2(D)$ mit $D \equiv [0, 1[$ und eine feste *Auflösungsstufe* $r, r \geq 0$. Gesucht sei eine gute Approximation von f in dem *Approximationsraum* $V^r \subset \mathcal{L}^2(D)$. V^j bezeichne dabei für $j \geq 0$ den Hilbert-Raum aller stückweise konstanten Funktionen über D , die höchstens an den Vielfachen von 2^{-j} eine Unstetigkeit aufweisen ($\dim V^j = 2^j$).

Das Problem kann in bezug auf die 2-Norm optimal durch die Orthogonalprojektion von f auf eine Basis von V^r gelöst werden. Der Einsatz einer Wavelet-Basis hat gegenüber dem Einsatz der kanonischen Basis bestehend aus Box-Funktionen den großen Vorteil, daß (zumindest teilweise) „glatte“ Funktionen durch „viele kleine“ Koeffizienten approximiert werden. Das Ignorieren derartiger Koeffizienten führt zu einer *kompakten* Darstellung einer *nahezu optimalen*¹⁴ Approximation.

Die *kanonische Basis* des Approximationsraums V^j ist wie folgt definiert (vgl. Abbildung 2.8):

$$\mathcal{K}_{V^j} \equiv \{\phi_i^j(x) \mid 0 \leq i < 2^j\} \quad \text{mit} \quad \phi_i^j(x) \equiv \phi(2^j x - i)$$

$$\text{und} \quad \phi(x) \equiv \begin{cases} 1, & x \in D \\ 0, & \text{sonst} \end{cases}$$

¹⁴ Der quadratische Fehler in bezug auf die 2-Norm entspricht – bei Verwendung einer Orthonormalbasis – gerade der Summe der Quadrate aller ignorierten Koeffizienten [SDS95].

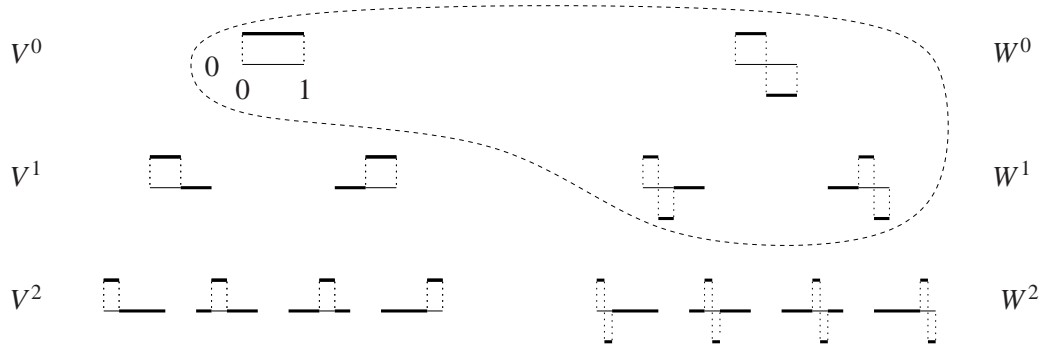


Abbildung 2.8 Kanonische Basen von Approximations- und Komplementräumen der ersten Auflösungsstufen (1D). Die Haar-Basis von V^2 ist durch die gestrichelte Linie gekennzeichnet.

Die Basisfunktionen werden *Skalierungsfunktionen* genannt. Alle Skalierungsfunktionen ϕ_i^j ergeben sich durch (Parameter-)Skalierung und (Parameter-)Translation ausgehend von der Mutter-Skalierungsfunktion ϕ . Der obere Index j gibt die Auflösungsstufe an und der untere Index i selektiert den Träger. Auf der größten Auflösungsstufe 0 befindet sich genau eine Basisfunktion – nämlich die Mutter-Skalierungsfunktion. Zu jeder Skalierungsfunktion existieren auf der nächstfeineren Auflösungsstufe genau zwei Skalierungsfunktionen, deren Träger gerade mit den Trägerhälften der ursprünglichen Basisfunktion übereinstimmen¹⁵. Folglich haben alle Skalierungsfunktionen einer Auflösungsstufe dieselbe Trägergröße, so daß der obere Index j auch die Skalierung angibt. Ein Skalierungsfunktion-Koeffizient korrespondiert zu dem mittleren Funktionswert von f über dem Träger. Die kanonische Basis ist orthogonal. Durch die Skalierung mit $2^{\frac{j}{2}}$ ergibt sich eine Orthonormalbasis.

Die folgenden Unterräume haben eine zentrale Bedeutung für die Konstruktion einer Wavelet-Basis eines Approximationsraums:

$$W^j \equiv \{g \in V^{j+1} \mid g \perp h \quad \forall h \in V^j\} \subset V^{j+1}$$

Da $V^j \subset V^{j+1}$ gilt, ist W^j der (*orthogonale*) *Komplementraum* von V^j in V^{j+1} . Informell enthält W^j – neben der Nullfunktion – diejenigen Funktionen, die für die Erweiterung der Funktionen aus V^j zu Funktionen aus V^{j+1} benötigt werden. Man schreibt:

$$V^{j+1} = V^j \oplus W^j \tag{2.7}$$

Insbesondere läßt sich jede Basis von V^j durch jede Basis von W^j zu einer Basis von V^{j+1} ergänzen. Die Dimension von W^j entspricht der Dimension von V^{j+1} abzüglich der Dimension von V^j ($\dim W^j = 2^j$).

Die *kanonische Basis* des Komplementraums W^j ist wie folgt definiert (vgl. Abbildung 2.8):

$$\mathcal{K}_{W^j} \equiv \{\psi_i^j(x) \mid 0 \leq i < 2^j\} \quad \text{mit} \quad \psi_i^j(x) \equiv \psi(2^j x - i)$$

$$\text{und} \quad \psi(x) \equiv \begin{cases} 1, & x \in [0, \frac{1}{2}[\\ -1, & x \in [\frac{1}{2}, 1[\\ 0, & \text{sonst} \end{cases}$$

Die Basisfunktionen werden *Wavelets* genannt. Im eindimensionalen Fall ergeben sich alle Wavelets ψ_i^j – analog zu den Skalierungsfunktionen – durch *Skalierung* und *Translation* ausgehend von

¹⁵ Auf diese Weise ist auf der Menge aller Skalierungsfunktionen eine *Eltern-Kind-Beziehung* definiert.

dem Mutter-Wavelet ψ . Zu jeder Skalierungsfunktion ϕ_i^j existiert genau ein korrespondierendes Wavelet ψ_i^j mit identischem Träger. Ein Wavelet-Koeffizient korrespondiert zu der Abweichung des mittleren Funktionswertes von f über der linken (bzw. rechten) Trägerhälfte von dem mittleren Funktionswert von f über dem gesamten Träger. Diese Basis ist ebenfalls orthogonal bzw. nach der Skalierung mit $2^{\frac{j}{2}}$ sogar orthonormal.

Die *Haar-Basis* des Approximationsraums V^j ist definiert als die Vereinigung der kanonischen Basen aller zueinander orthogonalen Unterräume, die sich durch die Auflösung der Rekursion in Gleichung 2.7 ergeben:

$$\mathcal{H}_{V^j} \equiv \mathcal{K}_{V^0} \cup \bigcup_{j'=0}^{j-1} \mathcal{K}_{W^{j'}} \quad (2.8)$$

In der Abbildung 2.8 ist die Haar-Basis von V^2 durch eine gestrichelte Linie gekennzeichnet. Die Haar-Basis hat einen *hierarchischen* Aufbau: Sie besteht aus genau einer Skalierungsfunktion auf der obersten Auflösungsstufe und „einer Pyramide“ von Wavelets auf verschiedenen Auflösungsstufen. Die Eltern-Kind-Beziehung auf der Menge aller Wavelets induziert eine Eltern-Kind-Beziehung auf der Menge aller Basisfunktionen der Haar-Basis, wobei das Mutter-Wavelet als Kind der Mutter-Skalierungsfunktion definiert ist. Der Skalierungsfunktion-Koeffizient korrespondiert zu dem mittleren Funktionswert von f über dem gesamten Definitionsbereich D . Ein Wavelet-Koeffizient kodiert die zu dem korrespondierenden Träger gehörige *Detailinformation*. Die Haar-Basis ist orthogonal bzw. – wenn sich die Konstruktion auf normierte kanonische Basen stützt – sogar orthonormal.

Kompakte Funktionsapproximation

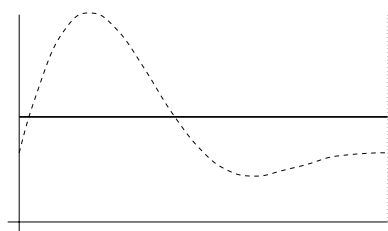
Die Schlüsseleigenschaft ist eine Folge der Orthogonalität von Wavelets zu Polynomen niedrigen Grades (*vanishing-moment-property*)¹⁶. Eine Funktion $g \in \mathcal{L}^2(D)$ hat genau dann m *verschwindende Momente*, wenn g zu allen Monomen $x \mapsto x^d$ vom Grad d mit $0 \leq d < m$ orthogonal ist¹⁷. Im Rahmen der kompakten Funktionsapproximation werden folglich diejenigen Funktionen als „glatt“ eingestuft, die „gut“ durch ein Polynom vom Grad $< m$ approximiert werden können. Die Anzahl der verschwindenden Momente der Wavelets einer Wavelet-Basis ist daher ein Maß für das Kompressions-Potential einer Wavelet-Basis. Die Haar-Wavelets haben „nur“ ein verschwindendes Moment, da sie zwar orthogonal zu allen konstanten Funktionen, aber nicht zu allen linearen Funktionen sind. In diesem Fall werden lediglich nahezu konstante Teilfunktionen von f als „glatt“ eingestuft und somit durch „kleine“ Wavelet-Koeffizienten approximiert.

Abbildung 2.9 veranschaulicht die Wavelet-basierte kompakte Funktionsapproximation anhand eines Beispiels. Abbildung (i) zeigt die optimale Approximation von f in dem Approximationsraum V^4 bzgl. der kanonischen Basis. Die Abbildungen (a), (b), (d), (f) und (h) zeigen dieselbe Approximation bzgl. der Haar-Basis. Die optimale Approximation ist jeweils durch 16 Koeffizienten eindeutig definiert. Ignoriert man die acht betragsmäßig kleinsten Wavelet-Koeffizienten¹⁸, so ergibt sich eine kompakte nahezu optimale Approximation. Abbildung (j) zeigt diese Approximation bzgl. der kanonischen Basis. Unsignifikante Details in „glatten“ Bereichen gehen verloren. Signifikante Details in Bereichen mit großen Gradienten bleiben erhalten.

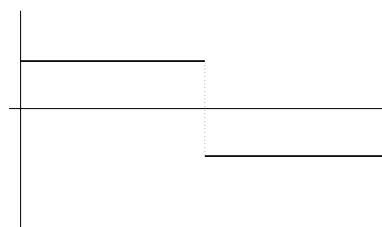
¹⁶ Ein formaler Beweis der Schlüsseleigenschaft mit Hilfe der Taylor'schen Formel ist z. B. in [Sch94b] dokumentiert.

¹⁷ äquivalent: ..., wenn die Orthogonalprojektion derartiger Monome auf g die Nullfunktion ergibt.

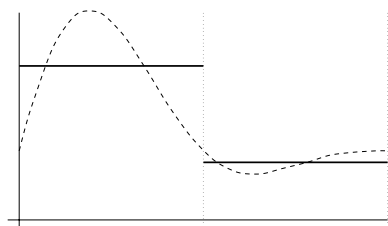
¹⁸ Die korrespondierenden Wavelets sind in der Abbildung 2.9 dünner dargestellt.



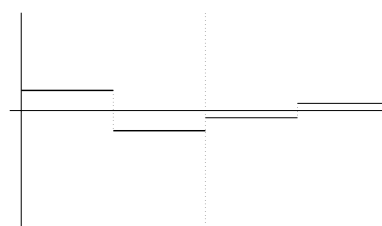
(a) V^0 -Approximation



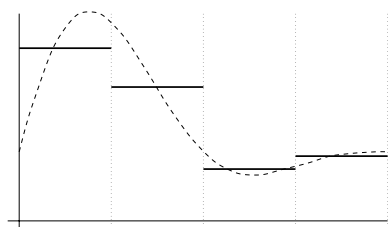
(b) W^0 -Approximation



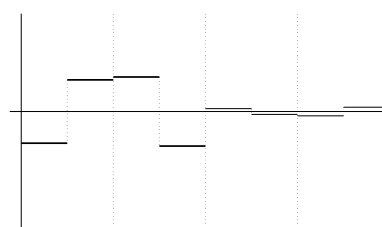
(c) V^1 -Approximation



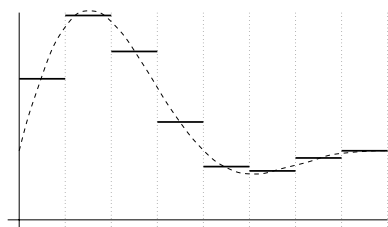
(d) W^1 -Approximation



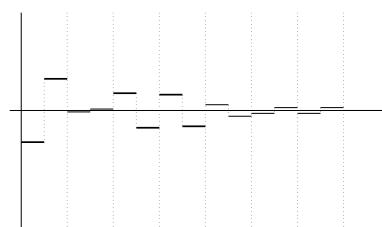
(e) V^2 -Approximation



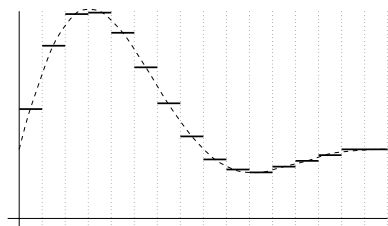
(f) W^2 -Approximation



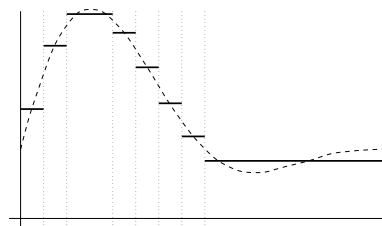
(g) V^3 -Approximation



(h) W^3 -Approximation



(i) V^4 -Approximation (16 Koeffizienten)



(j) kompakte V^4 -Approximation (8 Koeffizienten)

Abbildung 2.9 Wavelet-basierte kompakte Approximation von $f(x) \equiv 0.3 + (1 - x)^2 \sin(2\pi x)$

Effiziente Basistransformation

Jede Skalierungsfunktion und jedes Wavelet lässt sich als Linearkombination von Skalierungsfunktionen der nächstfeineren Auflösungsstufe ausdrücken und jede Skalierungsfunktion lässt sich umgekehrt als Linearkombination von Skalierungsfunktionen und Wavelets der nächstgrößeren Auflösungsstufe ausdrücken. Diese *two-scale-relationship* folgt aus Gleichung 2.7. Insbesondere sind aufgrund der Konstruktion von Wavelet-Basen jeweils nur wenige Basisfunktionen in die Beziehung involviert. Die Beziehung ist ferner translations- und skalierungsinvariant. Die two-scale-relationship für die unnormierte Haar-Basis ist vollständig durch die folgende 2×2 -Matrix definiert¹⁹:

$$\mathbf{W} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Für alle $j \geq 0$ und $i \in \{0, \dots, 2^j - 1\}$ gilt:

$$\begin{pmatrix} \phi_i^j \\ \psi_i^j \end{pmatrix} = \mathbf{W} \begin{pmatrix} \phi_{2i}^{j+1} \\ \phi_{2i+1}^{j+1} \end{pmatrix} \quad \text{bzw.} \quad \begin{pmatrix} \phi_{2i}^{j+1} \\ \phi_{2i+1}^{j+1} \end{pmatrix} = \mathbf{W}^{-1} \begin{pmatrix} \phi_i^j \\ \psi_i^j \end{pmatrix}$$

Weiter gilt für die korrespondierenden Koeffizienten [Gor95]:

$$\begin{pmatrix} c_{\phi_i^j} \\ c_{\psi_i^j} \end{pmatrix} = \mathbf{W}^{-T} \begin{pmatrix} c_{\phi_{2i}^{j+1}} \\ c_{\phi_{2i+1}^{j+1}} \end{pmatrix} \quad \text{bzw.} \quad \begin{pmatrix} c_{\phi_{2i}^{j+1}} \\ c_{\phi_{2i+1}^{j+1}} \end{pmatrix} = \mathbf{W}^T \begin{pmatrix} c_{\phi_i^j} \\ c_{\psi_i^j} \end{pmatrix}$$

Die two-scale-relationship ermöglicht ein effizientes „Navigieren“ zwischen benachbarten Auflösungsstufen. Insbesondere lassen sich unter Ausnutzung dieser Beziehung die Darstellungen einer Funktion bzgl. der kanonischen Basis und bzgl. einer Wavelet-Basis durch einen *Pyramidalalgorithmus* effizient ineinander überführen.

Die Transformation von der kanonischen Basis in eine Wavelet-Basis erfolgt durch eine *bottom-up* Traversierung der Koeffizienten-Pyramide mittels \mathbf{W}^{-T} . Die in einer Stufe berechneten Wavelet-Koeffizienten gehören zur Ausgabe und die berechneten Skalierungsfunktion-Koeffizienten dienen als Eingabe für die nächste Stufe. Der Koeffizient der Mutter-Skalierungsfunktion gehört ebenfalls zur Ausgabe. Da sich die Problemgröße mit jeder Stufe halbiert, ergibt sich insgesamt nur eine *lineare* Zeitkomplexität. Eine klassische Basistransformation durch Anwendung einer Transformationsmatrix hat demgegenüber eine quadratische Zeitkomplexität. Die inverse Transformation von einer Wavelet-Basis in die kanonische Basis erfolgt durch eine *top-down* Traversierung der Koeffizienten-Pyramide mittels \mathbf{W}^T . Die in einer Stufe berechneten Skalierungsfunktion-Koeffizienten dienen als Eingabe für die nächste Stufe. Die in der untersten Stufe berechneten Skalierungsfunktion-Koeffizienten bilden die Ausgabe. Die Zeitkomplexität ist ebenfalls linear.

2.2.4.3 Mehrdimensionale Haar-Basis

Viele der im eindimensionalen Fall eingeführten Notationen, Konstruktionen und Eigenschaften lassen sich auf den mehrdimensionalen Fall übertragen. Im folgenden werden nur die modifizierten und zusätzlichen Aspekte beschrieben. Die zu approximierende Funktion $f \in \mathcal{L}^2(D)$ habe nun den Definitionsbereich $D \equiv [0, 1]^d$ mit $d > 1$. Der Approximationsraum V_{dD}^j bestehe aus allen stückweise konstanten Funktionen über D , die *pro Dimension* höchstens an den Vielfachen von 2^{-j} eine Unstetigkeit aufweisen ($\dim V_{dD}^j = 2^{dj}$). Ein gebräuchliches Prinzip zur Konstruktion von mehrdimensionalen Basen besteht darin, eindimensionale Basisfunktionen durch *Tensorproduktbildung* zu mehrdimensionalen Basisfunktionen zu kombinieren.

¹⁹ Im normierten Fall ist die Beziehung analog durch \mathbf{W}' mit $\mathbf{W}' \equiv \frac{1}{\sqrt{2}}\mathbf{W}$ definiert.

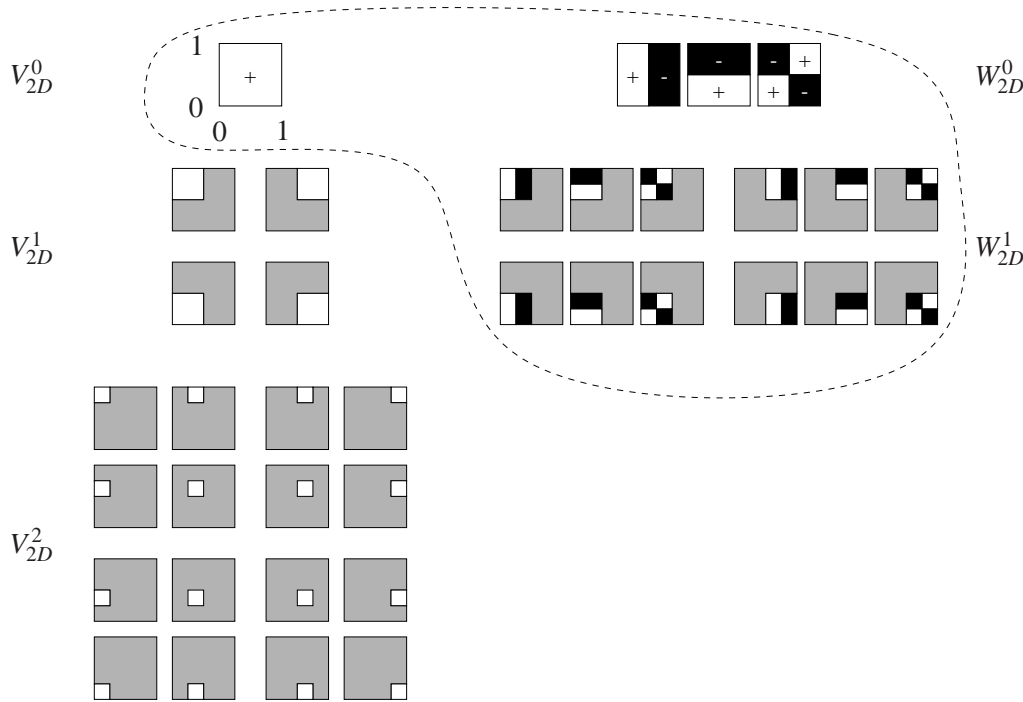


Abbildung 2.10 Kanonische Basen von Approximations- und Komplementräumen (Nicht-Standard-Konstruktion) der ersten Auflösungsstufen (2D). Weiße/graue/schwarze Bereiche korrespondieren zu den Funktionswerten $+1/0/-1$ (ohne Normierung). Die Nicht-Standard-Haar-Basis von V_{2D}^2 ist durch die gestrichelte Linie gekennzeichnet.

Die kanonische Basis des Approximationsraums V_{dD}^j ist wie folgt definiert (vgl. Abb. 2.10):

$$\begin{aligned} \mathcal{K}_{V_{dD}^j} &\equiv \{ \phi \dots \phi_{(i_1, \dots, i_d)}^j(x_1, \dots, x_d) \mid 0 \leq i_1, \dots, i_d < 2^j \} \quad \text{mit} \\ \phi \dots \phi_{(i_1, \dots, i_d)}^j(x_1, \dots, x_d) &\equiv \phi_{i_1}^j(x_1) \cdot \dots \cdot \phi_{i_d}^j(x_d) \end{aligned}$$

Alle Skalierungsfunktionen $\phi \dots \phi_{(i_1, \dots, i_d)}^j$ ergeben sich durch (simultane Parameter-)Skalierung und (Parameter-)Translation ausgehend von der Mutter-Skalierungsfunktion $\phi \dots \phi$. Der Träger wird durch einen Multiindex (i_1, \dots, i_d) selektiert. Zu jeder Skalierungsfunktion existieren auf der nächstfeineren Auflösungsstufe genau 2^d Skalierungsfunktionen, deren Träger gerade mit den Teilträgern übereinstimmen, die sich durch eine Halbierung des Trägers der ursprünglichen Skalierungsfunktion *pro Dimension* ergeben²⁰. Die Basis ist orthogonal. Wenn bei der Tensorproduktbildung normierte eindimensionale Skalierungsfunktionen verwendet werden, ist die Basis sogar orthonormal.

Im mehrdimensionalen Fall gibt es *zwei* gebräuchliche kanonische Basen für die orthogonalen Komplementräume W_{dD}^j von V_{dD}^j in V_{dD}^{j+1} ($\dim W_{dD}^j = (2^d - 1)2^{dj}$):

$$\mathcal{K}_{W_{dD}^j} \equiv \begin{cases} \{b_1 \dots b_d \mid \forall i : b_i \in \mathcal{H}_{V^j} \cup \mathcal{K}_{W^j}, \exists i' : b_{i'} \in \mathcal{K}_{W^j}\}, & \text{Standard-Konstruktion} \\ \{b_1 \dots b_d \mid \forall i : b_i \in \mathcal{K}_{V^j} \cup \mathcal{K}_{W^j}, \exists i' : b_{i'} \in \mathcal{K}_{W^j}\}, & \text{Nicht-Standard-Konstruktion} \end{cases}$$

Da die Konstruktion von mehrdimensionalen Haar-Basen analog zum eindimensionalen Fall aufbauend auf einer kanonischen Basis für die Komplementräume erfolgt (vgl. Definition 2.8), existieren im mehrdimensionalen Fall zwei gebräuchliche Haar-Basen: die *Standard-Haar-Basis* und

²⁰ Eltern-Kind-Beziehung auf der Menge aller Skalierungsfunktionen

die *Nicht-Standard-Haar-Basis*. Ausgehend von der eindimensionalen Haar-Basis besteht die Standard-Haar-Basis gerade aus allen möglichen Tensorprodukten. Insbesondere werden dabei Basisfunktionen *verschiedener* Auflösungsstufen miteinander kombiniert. Demgegenüber werden bei der Nicht-Standard-Haar-Basis unter Hinzunahme von Skalierungsfunktionen nur Funktionen *derselben* Auflösungsstufe miteinander kombiniert, wobei die ausschließliche Kombination von Skalierungsfunktionen nur auf der größten Auflösungsstufe erlaubt ist. Folglich gilt die folgende Aussage nur im Nicht-Standard-Fall: Alle Wavelets einer Auflösungsstufe haben *pro Dimension* dieselbe Trägergröße.

Die Nicht-Standard-Haar-Basis kann als die „natürlichere“ Erweiterung der eindimensionalen Haar-Basis angesehen werden (vgl. Abbildung 2.10): Alle Wavelets ergeben sich durch (simultane Parameter-)Skalierung und (Parameter-)Translation ausgehend von $2^d - 1$ Mutter-Wavelets $\phi \dots \phi \psi, \dots, \psi \dots \psi \psi$. Zu jeder Skalierungsfunktion $\phi \dots \phi_{(i_1, \dots, i_d)}^j$ existiert von *jedem* dieser Typen t genau ein Wavelet $t_{(i_1, \dots, i_d)}^j$ mit identischem Träger. Die Eltern-Kind-Beziehung auf der Menge aller Wavelets gleichen Typs induziert eine Eltern-Kind-Beziehung auf der Menge aller Basisfunktionen der Haar-Basis, wobei die Mutter-Wavelets als Kinder der Mutter-Skalierungsfunktion definiert sind.

2.2.4.4 Standard-Zerlegung des Transportoperators

Im Rahmen der Diskretisierung der Radiance-Gleichung wird die *Nicht-Standard-Haar-Basis* zur Approximation der mehrdimensionalen²¹ Radiance-Verteilung eingesetzt. Die Nicht-Standard-Konstruktion wird der Standard-Konstruktion hauptsächlich deswegen vorgezogen, weil die korrespondierenden Radiance-Koeffizienten aufgrund der „natürlicheren“ Erweiterung in einfachen Baum-Datenstrukturen verwaltet werden können (vgl. Abschnitte 4.1.1.1 bzw. 3.1.1.1).

Die Transportkoeffizienten lassen sich in diesem Fall als Koeffizienten des in eine *induzierte Standard-Haar-Basis* projizierten mehrdimensionalen²² Kernels interpretieren:

$$T_{r \leftarrow s} = \langle k, B_r \otimes B_s \rangle$$

Im Unterschied zu der Beschreibung im Abschnitt 2.2.4.3 liegt der Tensorprodukt-Konstruktion nicht die eindimensionale Haar-Basis zugrunde, sondern die mehrdimensionale Nicht-Standard-Haar-Basis. Da der Kernel implizit bzgl. einer Standard-Haar-Basis approximiert wird, spricht man auch von der *Standard-Zerlegung* des Transportoperators. Bei der Standard-Zerlegung können *alle* Radiance-Basisfunktionen über die Auflösungsstufen hinweg miteinander interagieren.

Bei der alternativen *Nicht-Standard-Zerlegung* des Transportoperators wird der Kernel bzgl. der Nicht-Standard-Haar-Basis approximiert. Die Nicht-Standard-Zerlegung setzt die Approximation der Radiance-Verteilung bzgl. eines linear abhängigen *Erzeugendensystems* bestehend aus der Nicht-Standard-Haar-Basis *und* den korrespondierenden Skalierungsfunktionen voraus. Diese „Über-Repräsentation“ der Radiance-Verteilung ist charakteristisch für eine Nicht-Standard-Zerlegung. In diesem Fall können nur Radiance-Funktionen des Erzeugendensystems *derselben* Auflösungsstufe miteinander interagieren, wobei die Interaktion von Skalierungsfunktionen

²¹ vierdimensionalen (Radiance-Fall) bzw. zweidimensionalen (Radiosity-Fall)

²² achtdimensionalen (Radiance-Fall) bzw. vierdimensionalen (Radiosity-Fall) – der im Radiance-Fall gemäß der Def. in Abschnitt 2.2 sechsdimensionale Kernel wird wie folgt um einen zusätzlichen Richtungsparameter $\vec{\omega}'$ erweitert:

$$k(x, \vec{\omega}, x', \vec{\omega}') \equiv \begin{cases} k(x, \vec{\omega}, x'), & \vec{\omega}' = x - x' \\ 0, & \text{sonst} \end{cases}$$

nur auf der größten Auflösungsstufe erlaubt ist. Die Darstellungen einer Funktion bzgl. der kanonischen Basis und bzgl. dieses Erzeugendensystems lassen sich durch einen leicht *modifizierten* Pyramidenalgorithmus effizient ineinander überführen. Die Modifikation beschränkt sich auf die zusätzliche Berücksichtigung der Skalierungsfunktion-Koeffizienten in jeder Stufe. Da die Radiance-Verteilung bei der Nicht-Standard-Zerlegung „über-repräsentiert“ ist, muß die Darstellung nach jeder Anwendung der Transportmatrix in eine *konsistente* Form gebracht werden. Dieses Problem läßt sich effizient durch den folgenden zweistufigen Ansatz lösen. Zunächst wird die „Über-Repräsentation“ mittels eines modifizierten top-down Pyramidenalgorithmus in eine Darstellung bzgl. der kanonischen Basis überführt (*Push*). Danach wird diese Darstellung mittels eines modifizierten bottom-up Pyramidenalgorithmus in eine konsistente „Über-Repräsentation“ überführt (*Pull*).

Der Einsatz von Wavelet-Basen zur effizienten Anwendung von Integraloperatoren auf Funktionen basiert auf [BCR91]. Beylkin hat bewiesen, daß eine große Klasse von Integraloperatoren, deren Kernel gewisse allgemeine Glätte-Kriterien erfüllen, durch Matrizen bestehend aus $O(n \log n)$ (Standard-Zerlegung) bzw. $O(n)$ (Nicht-Standard-Zerlegung) signifikanten Einträgen bei einer vorgegebenen Fehlertoleranz approximiert werden können. Die korrespondierenden Konstanten hängen von dem Kompressions-Potential der eingesetzten Wavelet-Basis ab. Beide Zerlegungen sind der Projektion des Kerns in eine nicht-hierarchische Basis mit ihren n^2 Koeffizienten deutlich überlegen. Da der Transportkernel – bei Ignorierung der Sichtbarkeitsfunktion – die o. g. Glätte-Kriterien erfüllt²³[Sch94b], lassen sich Beylkins Ergebnisse auf das Lösen der Radiance-Integralgleichung übertragen. Christensen verfolgt trotz der etwas schlechteren asymptotischen Komplexität die Standard-Zerlegung des Transportoperators, da diese nach den praktischen Erfahrungen verschiedener Autoren häufig zu einer dünner besetzten Transportmatrix führt.

2.2.4.5 Auswahl einer Wavelet-Basis

Die Haar-Basis ist „nur“ der einfachste Vertreter einer großen Klasse von Wavelet-Basen. Im Rahmen der Diskretisierung der Radiance-Gleichung sind alternativ u. a. Flatlets bzw. Multiwavelets [GSCH93] [Sch94b] und Coiflets bzw. Splinelets [Chr95] eingesetzt worden. Bei der Auswahl einer Wavelet-Basis gilt es, diverse wünschenswerte Eigenschaften gegeneinander abzuwägen.

Beispielsweise besteht ein enger Zusammenhang zwischen dem Kompressions-Potential und dem Quadratur-Aufwand: Einerseits weist die Haar-Basis im Vergleich zu den o. g. alternativen Wavelet-Basen nur ein geringes Kompressions-Potential auf. Andererseits lassen sich die einzelnen Transportkoeffizienten vergleichsweise effizient berechnen. Eine Erhöhung des Kompressions-Potentials ist mit einer Verbreiterung der Träger der Basisfunktionen und/oder mit einer Erhöhung der polynomiellen Ordnung der Basisfunktionen verbunden. In beiden Fällen erhöht sich der Aufwand für die Berechnung eines einzelnen Transportkoeffizienten und der Aufwand für die Abschätzung seiner Signifikanz.

Ein anderer Aspekt ist die Stetigkeit der Basisfunktionen. Eine bzgl. einer Wavelet-Basis höherer polynomieller Ordnung über einem Ausgangs-Patch repräsentierte Radiance-Verteilung eignet sich direkt für die Ausgabe. Demgegenüber weist eine bzgl. der Haar-Basis repräsentierte Radiance-Verteilung an den Grenzen der Teilträger i. a. visuell störende Helligkeitssprünge auf. In diesem Fall ist eine „Glättung“ der berechneten Radiance-Verteilung bei der graphischen Ausgabe durch eine *Rekonstruktionsmethode* – z. B. Gouraud-Shading oder Final-Gathering – erforderlich.

Die Dimension von V^0 sollte möglichst gering sein. Im Radiance-Fall gilt: b Skalierungsfunktionen zum Aufspannen von V^0 implizieren b^4 Skalierungsfunktionen zur Darstellung der vierdimensionalen Radiance-Verteilung über einem Ausgangs-Patch auf der größten Auflösungsstufe

²³ Die Argumentation basiert u. a. auf der quadratischen Abnahme des Kerns mit der Entfernung zwischen zwei Oberflächenpunkten.

bzw. b^8 Skalierungsfunktionen zur Darstellung des Transports zwischen zwei Ausgangs-Patches auf der größten Auflösungsstufe. Die Haar-Basis ist diesbezüglich mit $b = 1$ optimal. Ferner sollten sich die Translierten einer Skalierungsfunktion bzw. die Translierten eines Wavelets innerhalb einer Auflösungsstufe nicht überlappen. Dadurch vereinfachen sich die Datenstrukturen zur Verwaltung der Radiance-Koeffizienten. Eine weitere sehr wünschenswerte Eigenschaft ist die Orthonormalität. Orthonormalbasen sind insbesondere selbst-dual. Die Haar-Basis erfüllt die beiden letztgenannten Eigenschaften.

Christensen hat den Einsatz von Haar-Wavelets, Coiflets und Splinelets untersucht. Aufgrund seiner experimentellen Ergebnisse und nach Abwägung der o. g. Aspekte hat er sich für die Verwendung der Haar-Basis entschieden.

2.3 Basis-Algorithmus

In diesem Abschnitt wird das Grundgerüst eines Wavelet-basierten globalen Beleuchtungsalgorithmus vorgestellt. Als Eingabe erwartet der Algorithmus eine statische synthetische Szene bestehend aus einer Menge von dreidimensionalen geometrischen Objekten mit wohldefinierten optischen Oberflächeneigenschaften²⁴. Der Algorithmus berechnet zunächst eine Approximation der resultierenden Radiance-Verteilung auf Basis des im Abschnitt 2.1 vorgestellten physikalisch basierten Beleuchtungsmodells mit Hilfe der im Abschnitt 2.2 eingeführten mathematischen Werkzeuge. Insbesondere wird die Standard-Zerlegung des Transportoperators verfolgt, wobei die Radiance-Verteilung explizit bzgl. Nicht-Standard-Haar-Basen repräsentiert wird. Danach erfolgt die graphische Ausgabe der berechneten Radiance-Verteilung.

Die Berechnung umfaßt i. w. die folgenden beiden fundamentalen Operationen:

Aufstellen der Transportmatrix bzw. Berechnung der signifikanten Transportkoeffizienten

Lösen der diskreten Radiance-Gleichung bzw. Durchführung von Matrix-Vektor-Multiplikationen (Gauss-Seidel-Iterationen)

Die spezielle Charakteristik des globalen Beleuchtungsproblems beeinflusst das Design des Algorithmus entscheidend. Im Vergleich zu anderen verwandten Anwendungsproblemen wird der gesamte Berechnungsaufwand von dem Aufwand für das Aufstellen der Matrix dominiert. Ein Transportkoeffizient ist als mehrdimensionales Integral definiert. Zudem hängt jeder Kernel-Wert bedingt durch die Sichtbarkeitsfunktion von der kompletten Szenengeometrie ab. Demgegenüber konvergiert das Gauss-Seidel-Verfahren numerisch i. a. bereits nach wenigen Iterationen, da das physikalische System i. a. stark gedämpft ist und der Kernel quadratisch mit der Entfernung zwischen zwei Oberflächenpunkten abnimmt [Sch96b]. Der Zeit- und Speicheraufwand der Berechnung ist proportional zu der Anzahl der zu berechnenden Transportkoeffizienten.

Das Aufstellen der Transportmatrix könnte durch einen einfachen *bottom-up* Ansatz erfolgen. Zunächst werden die zu allen geordneten Paaren von n Skalierungsfunktionen einer feinsten Auflösungsstufe korrespondierenden n^2 Transportkoeffizienten berechnet. Danach werden die Koeffizienten durch einen *bottom-up* Pyramidenalgorithmus in Koeffizienten bzgl. der Standard-Haar-Basis transformiert. Schließlich werden die unsignifikanten Koeffizienten entfernt. Dieser Ansatz ist jedoch nicht praktikabel. Aufgrund der o. g. Charakteristik des globalen Beleuchtungsproblems amortisieren sich die Kosten für das Aufstellen der Transportmatrix nicht durch die wenigen effizienten Matrix-Vektor-Multiplikationen. Zeit- und Speicheraufwand betragen $O(n^2)$.

Statt dessen wird ein effizienterer und flexiblerer *top-down* Ansatz verfolgt. Die Grundidee besteht darin, nur die wenigen signifikanten Transportkoeffizienten im Rahmen eines *rekursiven*

²⁴ Emissions- und Reflexionseigenschaften

Aufzählungsschemas mit Hilfe eines Orakels effizient aufzuspüren und direkt zu berechnen. Der folgende Pseudo-Code beschreibt den Basis-Algorithmus:

```

void GlobalIllumination(Le) {
  T ← {Tr←s | Br, Bs ∈ KVdiff0} // initial linking
  L ← Solve(T, Le)
  while (quality of solution is not sufficient) {
    T ← Refine(T, L)
    L ← Solve(T, Le)
  }
  L' ← PyramidDown(L)
  Render(L')
}

```

Die Oberflächen der geometrischen Objekte seien jeweils durch ein Netz von planaren Ausgangs-Patches approximiert. Die Radiance-Verteilung wird über jedem der insgesamt k Ausgangs-Patches bzgl. einer dynamisch wachsenden Nicht-Standard-Haar-Basis repräsentiert. Anfangs wird jeweils nur der Koeffizient der Mutter-Skalierungsfunktion verwaltet. Die gegebene emittierende Radiance-Verteilung sei bzgl. der Mutter-Skalierungsfunktionen definiert. Insbesondere strahlen die Lichtquellen ihre Energie – auch im Radiance-Fall – diffus aus. Die Transportmatrix umfaßt anfangs die zu allen geordneten Paaren von Mutter-Skalierungsfunktionen korrespondierenden k^2 Transportkoeffizienten (*initial linking*). Die so definierte „grobe“ diskrete Radiance-Gleichung wird nach dem Gauss-Seidel-Verfahren gelöst und man erhält eine „grobe“ – auch im Radiance-Fall richtungsunabhängige – Radiance-Verteilung. Im Hauptteil des Algorithmus wird alternierend die aktuelle diskrete Radiance-Gleichung verfeinert und eine genauere Radiance-Verteilung durch das Lösen des verfeinerten Gleichungssystems berechnet. Die Schleife terminiert, wenn die aktuelle Radiance-Verteilung nach Einschätzung des Anwenders eine ausreichende Qualität aufweist. Vor der graphischen Ausgabe werden die bzgl. einer Nicht-Standard-Haar-Basis definierten Radiance-Koeffizienten durch einen top-down Pyramidenalgorithmus in bzgl. einer kanonischen Basis definierte Radiance-Koeffizienten transformiert. Die graphische Ausgabe der berechneten Radiance-Verteilung basiert auf den transformierten Koeffizienten. Der Zeit- und Speicheraufwand der Berechnung beträgt $O(k^2 + n \log n)$, wobei n die Anzahl der Skalierungsfunktionen auf den jeweils feinsten Auflösungsstufen bezeichnet.

Abschließend soll die Verfeinerung der aktuellen diskreten Radiance-Gleichung detaillierter beschrieben werden. Ziel ist die Erweiterung der Transportmatrix um wenige signifikante Transportkoeffizienten. Die erweiterte Transportmatrix approximiert den Transportoperator genauer. Folglich approximiert auch die unter Verwendung der erweiterten Transportmatrix berechnete Radiance-Verteilung die exakte Radiance-Verteilung genauer. In einem Verfeinerungsschritt werden die existierenden Transportkoeffizienten um *höchstens eine* Stufe entweder bzgl. der Empfänger-Basisfunktion oder bzgl. der Sender-Basisfunktion verfeinert. Dabei wird ggf.²⁵ auch die Darstellung der Radiance-Verteilung des Empfängers bzw. des Senders um eine Stufe verfeinert. Die Transportmatrix wird nur um solche Transportkoeffizienten erweitert, die von dem Orakel als signifikant eingestuft werden. Das Signifikanzmaß hängt auch von der aktuellen Radiance-Verteilung ab. Da die verfeinerten Transportkoeffizienten bei der verfolgten Standard-Zerlegung des Transportoperators *nicht* entfernt werden, muß ihre Genauigkeit nach dem Hinzufügen von feineren Transportkoeffizienten verbessert werden [Chr95] [Sch94b]. Christensen bezeichnet diesen Vorgang mit *adaptive numerische Integration*. Die adaptive numerische Integration ist ein notwendiger Bestandteil der Verfeinerungs-Prozedur.

²⁵ genau dann, wenn die entsprechende Stufe noch nicht zuvor im Rahmen der Verfeinerung eines anderen Transportkoeffizienten angelegt worden ist

Kapitel 3

Wavelet-Radiosity

In dem vorigen Kapitel wurde der theoretische Hintergrund von Wavelet-basierten globalen Beleuchtungsalgorithmen aufgearbeitet. Die Beschreibung endet mit der Vorstellung des Grundgerüsts eines Basis-Algorithmus. Die Implementierung des Wavelet-Radiance-Algorithmus erfolgte in zwei Phasen. Zunächst wurde ein Wavelet-Radiosity-Algorithmus implementiert. In dieser Phase lag der Schwerpunkt auf der Implementierung der ungewöhnlichen Standard-Zerlegung des Transportoperators. Danach wurde darauf aufbauend der Wavelet-Radiance-Algorithmus realisiert. In dieser Phase galt das Hauptaugenmerk der Implementierung der allgemeinen Reflexionseigenschaften. Da sich diese Vorgehensweise bewährt hat, folgt die weitere Beschreibung der vorwiegend praktischen Aspekte des Basis-Algorithmus demselben Prinzip. Dieses Kapitel beschränkt sich ausschließlich auf den Radiosity-Fall. Das nächste Kapitel bezieht sich auf den allgemeinen Radiance-Fall.

Zunächst wird die Implementierung der wesentlichen Komponenten des Basis-Algorithmus beschrieben. Es folgt ein Vergleich mit dem klassischen Hierarchical-Radiosity-Algorithmus. Anschließend wird die Anbindung der Implementierung an das 3D-Graphiksystem *Minimal Rendering Toolkit* (MRT) erläutert. Danach werden die im Rahmen dieser Arbeit verfolgten Erweiterungen des Basis-Algorithmus – einschließlich ihrer Implementierungen – vorgestellt. Schließlich werden die Ergebnisse von einigen Wavelet-Radiosity-Berechnungen dokumentiert.

3.1 Implementierung des Basis-Algorithmus

3.1.1 Datenstrukturen

3.1.1.1 Verwaltung der Radiance-Koeffizienten

Die zweidimensionale richtungsunabhängige Radiance-Verteilung wird über jedem Ausgangspatch bzgl. einer dynamisch wachsenden Nicht-Standard-Haar-Basis repräsentiert. Diese Darstellung lässt sich – mit Ausnahme des obersten Levels – durch eine einfache *quadtree*-Datenstruktur realisieren, die sich an der Eltern-Kind-Beziehung orientiert (vgl. Abbildung 2.10, Seite 26):

- Die Wurzel verwaltet den Koeffizienten der Mutter-Skalierungsfunktion. Die Wurzel hat entweder kein Kind oder genau ein Kind.
- Ein von der Wurzel verschiedener Knoten verwaltet die Koeffizienten von drei Wavelets mit identischem Träger. Ein derartiger Knoten hat entweder kein Kind oder genau vier Kinder.

Die implementierte Datenstruktur weicht wie folgt von dieser konzeptionellen Datenstruktur ab:

Verschmelzen der beiden obersten Level Die beiden obersten Level – jeweils ein Knoten – werden zu einem einzigen Knoten verschmolzen. Dadurch vereinfacht sich die Handhabung der Datenstruktur, da in diesem Fall eine quadtree-Datenstruktur *ohne* Ausnahme vorliegt. Des weitern werden konsequent alle Basisfunktionen mit identischem Träger in einem Knoten verwaltet.

Anhängen eines zusätzlichen Levels Zu jedem Blatt der konzeptionellen Datenstruktur wird ein zusätzlicher Level angelegt¹. Die neuen Blätter verwalten keine für die Definition der Radiance-Verteilung relevanten Koeffizienten. Das Anhängen eines zusätzlichen Levels ist primär wichtig für die graphische Ausgabe. In den neuen Blättern werden die von dem top-down Pyramidalalgorithmus berechneten Skalierungsfunktion-Koeffizienten abgelegt. Die graphische Ausgabe basiert auf diesen Koeffizienten. Ferner wird der zusätzliche Level auch bei der Berechnung der Transportkoeffizienten ausgenutzt, indem die Teilträger von Basisfunktionen separat „gesamlet“ werden.

Außerdem verwaltet jeder Knoten eine Liste von *Links*. Links werden im nächsten Abschnitt eingeführt. Jeder Knoten wird mit seinem als Träger dienenden Patch identifiziert. Die Patches einer Patch-Hierarchie sind als Instanzen einer Patch-Klasse implementiert. Der folgende Pseudo-Code faßt die wesentlichen von einem Patch verwalteten Daten noch einmal zusammen:

```
class t_WBPatch {
    // ===== coefficients of basisfunctions =====
    t_Color*   cPhi;           // pointer to coefficient of scaling-function
                                // - allocated <==> node==root or node==leaf
                                // - part of hierarchical representation only
                                //   for root nodes
    t_Color*   cPsi;           // pointer to coefficients of three wavelets
                                // - allocated <==> node!=leaf
    // ===== list of links =====
    t_WBLink*  links;          // node!=root and node==leaf ==> links==NULL
    // ===== hierarchy related data =====
    t_WBPatch* subPatches;     // pointer to four subpatches
};
```

Der Koeffizient der Mutter-Skalierungsfunktion entspricht der mittleren Radiance über dem Ausgangs-Patch. Als Patches werden Dreiecke und planare Vierecke unterstützt. Die vier Kinder eines Patches korrespondieren jeweils zu einem Viertel des ursprünglichen Patches mit identischem geometrischem Typ. Ein Skalierungsfunktion-Koeffizient entspricht der mittleren Radiance über dem Patch. Ein Wavelet-Koeffizient gibt die Abweichung der mittleren Radiance über zwei der vier Teil-Patches von der mittleren Radiance über dem gesamten Patch an. Es gibt genau drei Möglichkeiten, vier Elemente als ungeordnetes Paar bestehend aus ungeordneten Paaren zu organisieren. Intuitiv können die Wavelet-Koeffizienten als „horizontale“, „vertikale“ und „diagonale“ Abweichung interpretiert werden. Die Klasse zur Verwaltung von Radiance-Koeffizienten muß mit negativen Werten umgehen können, da die Wavelet-Koeffizienten negativ sein können.

3.1.1.2 Verwaltung der Transportkoeffizienten

Ein Transportkoeffizient gibt den Einfluß einer *Sender*-Basisfunktion auf eine *Empfänger*-Basisfunktion an (vgl. Seite 18). Transportkoeffizienten werden in *Links* zwischen dem zugehörigen Empfänger-Patch und dem zugehörigen Sender-Patch verwaltet. Links werden ihrerseits in einer dynamisch wachsenden Liste von ihrem Empfänger-Patch verwaltet. Im Rahmen dieser Arbeit

¹ Ausnahme: Die konzeptionelle Datenstruktur besteht nur aus der Wurzel.

zeigen Links nach Konvention – entgegengesetzt der Transportrichtung – von ihrem Empfänger-Patch zu ihrem Sender-Patch.

Um trotz der Verschmelzung der beiden obersten Level der konzeptionellen Datenstruktur zwischen Transporten von/zu der Mutter-Skalierungsfunktion und Transporten von/zu den Mutter-Wavelets unterscheiden zu können, werden vier verschiedene *Link-Typen* eingeführt: LT_SS, LT_WS, LT_SW und LT_WW. S steht für Mutter-Skalierungsfunktion und W steht für Wavelets. Beispielsweise werden in einem Link vom Typ LT_WS Transportkoeffizienten zwischen Wavelets des Empfänger-Patches und der Mutter-Skalierungsfunktion des Sender-Patches verwaltet. Das initial linking erfolgt ausschließlich über Links vom Typ LT_SS.

Die nächste Design-Entscheidung bezieht sich darauf, wieviele Transportkoeffizienten in einem Link verwaltet werden. Einerseits könnte man in einem Link *alle möglichen* Transportkoeffizienten verwalten, d. h. gemäß dem Typ einen, drei, drei bzw. neun Transportkoeffizienten. Ein Vorteil dieser Linking-Strategie besteht darin, daß die in einem Link verwalteten Transportkoeffizienten kostengünstig *simultan* berechnet werden können. Die Kosten für die simultane Berechnung derartiger Transportkoeffizienten entsprechen i. w. den Kosten für die herkömmliche Berechnung eines einzigen Transportkoeffizienten. Ein weiterer Vorteil ist der geringere Verwaltungs-Overhead, da vergleichsweise wenige Links zur Verwaltung der Transportkoeffizienten genügen. Andererseits könnte man in einem Link *genau einen* Transportkoeffizienten verwalten. Für diese Linking-Strategie spricht die höhere Flexibilität. Es werden ausschließlich signifikante Transportkoeffizienten und ihre Geschwister verwaltet. Demgegenüber werden bei der ersten Strategie i. a. auch unsignifikante Transportkoeffizienten in einem Link mitverwaltet. Anstelle dieser beiden extremen Strategien wird als Kompromiß die folgende Linking-Strategie verfolgt: In einem Link vom Typ LT_SS, LT_WS oder LT_SW werden alle möglichen Transportkoeffizienten verwaltet. In einem Link vom Typ LT_WW werden die drei Transportkoeffizienten zwischen *allen* Wavelets des Empfänger-Patches und *genau einem* Wavelet des Sender-Patches verwaltet². Folglich können zwischen einem geordneten Patch-Paar bis zu drei Links vom Typ LT_WW existieren. Jeder dieser Links zeigt auf ein anderes Wavelet des Sender-Patches.

Außerdem verwaltet jeder Link die maximale Farbkomponente der bei der Berechnung der Transportkoeffizienten ermittelten Variation des Kernels. Auf dieser Information basiert das Orakel. Links vom Typ LT_WS, LT_SW und LT_WW verwalten zusätzlich den Transportkoeffizienten zwischen den beiden lokalen Skalierungsfunktionen. Diese Transportkoeffizienten sind *nicht* Bestandteil der Transportmatrix. Sie werden für die Implementierung der adaptiven numerischen Integration benötigt. In dem folgenden Pseudo-Code sind die wesentlichen von einem Link verwalteten Daten noch einmal aufgeführt:

```
class t_WBLink {
    // ===== management =====
    t_WBPatch* senderPatch;    // pointer to sender
    t_LinkType type;          // LT_SS, LT_WS, LT_SW, OR LT_WW
    unsigned   waveletIndex;  // index of sending wavelet (0,1,2)
                                // - only for links of type LT_WW
    t_WBLink*  nextLink;     // pointer to next link in the list
    // ===== transportcoefficients (TCs) =====
    t_Color    cT;           // TC between (local) scaling-functions
                                // - part of transportmatrix only for
                                //   links of type LT_SS
    t_Color    cTArray[3];   // array of TCs between basisfunctions
                                // - only for links of type LT_WS,
                                //   LT_SW, or LT_WW
    // ===== other data =====
    t_Real     kernelVariation; // used by oracle
};
```

² Diese Transportkoeffizienten werden von dem Orakel als gleich signifikant eingeschätzt.

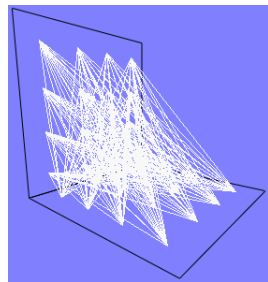
3.1.2 Berechnung der Transportkoeffizienten

Ein Transportkoeffizient ist als vierdimensionales Integral über dem Produkt des Kernels mit den beiden zugehörigen Basisfunktionen definiert (vgl. Seite 18). Die Grundidee von vielen numerischen Integrationsverfahren besteht darin, das zu berechnende Integral durch eine gewichtete Summe von Werten des Integranden über einer endlichen Menge von Stützstellen aus dem Integrationsbereich zu approximieren [Sto83]:

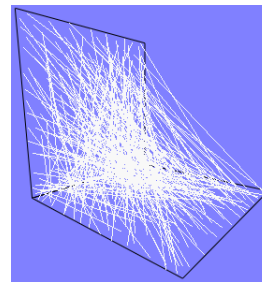
$$\int_I f(x) dx \approx \sum_{i=0}^{\#Stützstellen-1} w_i f(x_i) \quad \text{mit } x_i \in I$$

Transportkoeffizienten werden mittels einer sehr einfachen Variante berechnet: Die Stützstellen werden über dem Integrationsbereich möglichst gleichmäßig verteilt, und die korrespondierenden Werte des Integranden werden mit demselben Faktor $\frac{|I|}{\#Stützstellen}$ gewichtet.

Die numerische Quadratur wird auf den *relevanten* Integrationsbereich $supp(B_r) \times supp(B_s)$ beschränkt, d. h. Stützstellen korrespondieren zu Paaren bestehend aus einem Sample auf dem Empfänger-Patch und einem Sample auf dem Sender-Patch. Samples können auf einem Patch entweder *regulär* oder *gleichverteilt zufällig* platziert werden. Eine reguläre Platzierung wird jedoch nur für planare Parallelogramme tatsächlich durchgeführt. Andersförmige Patches werden stets gleichverteilt zufällig „gesamlet“. Samples auf dem Empfänger-Patch können entweder mit *allen* Samples auf dem Sender-Patch zu Stützstellen kombiniert werden oder mit *genau einem* Sample. Die letzte Variante läuft auf ein *N-Rooks-Sampling* hinaus [Shi90]. Abbildung 3.1 veranschaulicht zwei der vier möglichen Sampling-Strategien. Die voreingestellte Strategie (regular, one-to-all) hat z. B. den Vorteil, daß aus Symmetrie-Gründen identische Formfaktoren auch identisch approximiert werden und diesbezüglich keine Artefakte auftreten. Allerdings ist die Dichte der Samples auf den Patches vergleichsweise gering. Ferner kann die Anzahl der Stützstellen bei dieser Strategie nur in großen Schritten variiert werden.



(a) (regular, one-to-all)



(b) (random, one-to-one)

Abbildung 3.1 Sampling-Strategien (#Stützstellen = 256)

Alle Transportkoeffizienten werden mit derselben Anzahl von Stützstellen berechnet. Die Anzahl kann über eine Programm-Option vorgegeben werden. Für jede der vier Strategien werden jedoch nur bestimmte Werte unterstützt. Diese Einschränkung stellt sicher, daß auf jede der vier (LT_WS bzw. LT_SW) bzw. 16 (LT_WW) Kombinationen von Teilträgern des Empfänger-Patches mit Teilträgern des Sender-Patches dieselbe Anzahl von Stützstellen entfällt. Anstelle der *gewünschten* Anzahl von Stützstellen wird automatisch die am nächsten liegende *unterstützte* Anzahl von Stützstellen verwendet. Die unterstützten Werte sind in Abhängigkeit der Strategie und einem Parameter in der Tabelle 3.1 dokumentiert. Die Anzahl der auf jedem Patch platzierten

	one-to-all Sampling	one-to-one Sampling
regular Sampling	$16p^4$	$16p^2$
random Sampling	$16p^2$	$16p$

Tabelle 3.1 Unterstützte Anzahl von Stützstellen ($p > 0$)

Samples entspricht entweder der Quadratwurzel der Anzahl der Stützstellen (one-to-all Sampling) oder direkt der Anzahl der Stützstellen (one-to-one Sampling).

Die Multiplikation des Kernels mit den beiden zugehörigen Basisfunktionen ist bedingt durch den Einsatz der Haar-Basis sehr einfach. Der zu einer Stützstelle korrespondierende Kernel-Wert muß i. w. nur mit einem geeigneten Vorzeichen versehen werden. Das Vorzeichen hängt davon ab, in welche Teilträger die Samples fallen.

Die Berechnung eines Transportkoeffizienten ist vor allem deswegen sehr aufwendig, weil jeder Kernel-Wert bedingt durch die Sichtbarkeitsfunktion von der kompletten Szenengeometrie abhängt. Die Sichtbarkeit eines Sample-Paares wird durch Strahlschuß ermittelt. Zwei Samples sind genau dann gegenseitig sichtbar, wenn der Strahl von einem Sample in Richtung des anderen Sample zwischen den beiden Samples kein blockierendes Objekt schneidet (vgl. auch Abschnitt 3.3.2). Die Sichtbarkeit wird nach der Strategie *visibility-in-quadrature* berücksichtigt [WH97], d. h. bei der Berechnung eines Kernel-Wertes wird auch der zugehörige Sichtbarkeitswert ermittelt und direkt mit einbezogen. Alternativ könnte man die Sichtbarkeit eines Patch-Paares mit Hilfe eines zweiten Satzes von Strahlen ermitteln, und den *unverdeckten Transportkoeffizienten*³ mit dem Bruchteil der nicht-blockierten Strahlen gewichten. Die verwendete Strategie ist genauer, da die Kernel-Werte unter Berücksichtigung der zugehörigen Sichtbarkeitswerte akkumuliert werden. Das Orakel profitiert ebenfalls von dieser Vorgehensweise. Die alternative Strategie ist flexibler, da die Anzahl der Sichtbarkeitsstrahlen nicht mit der Anzahl der Stützstellen übereinstimmen braucht.

Die Sichtbarkeitsberechnung wird dadurch optimiert, daß *vollständige Sichtbarkeit* vererbt wird [HSA91]. Die Optimierung basiert auf der folgenden trivialen Beobachtung: Wenn ein Patch-Paar gegenseitig vollständig sichtbar ist, so ist insbesondere jedes Paar von zugehörigen Teil-Patches gegenseitig vollständig sichtbar. Falls bei der Berechnung eines Transportkoeffizienten vollständige Sichtbarkeit ermittelt wird⁴, unterbleibt die teure Ermittlung der Sichtbarkeit für alle Nachfahren. Diese populäre Optimierung kann jedoch zu Artefakten führen, falls ein Patch-Paar z. B. bedingt durch ein sehr grobes Sampling *fälschlicherweise* als gegenseitig vollständig sichtbar eingestuft wird. Diese Fehleinschätzung kann später nicht mehr revidiert werden. Aus diesem Grund läßt sich die Optimierung bei Bedarf deaktivieren.

Transportkoeffizienten werden bei der Erzeugung von Links berechnet. Links werden beim initial linking und im Rahmen der Verfeinerung erzeugt. Alle in einem Link verwalteten Transportkoeffizienten werden *simultan* berechnet, d. h. der Kernel wird nur einmal an derselben Menge von Stützstellen ausgewertet. Der folgende Pseudo-Code beschreibt das Grundgerüst der Berechnungs-Methode:

```
void t_WBLink::computeTCs(t_WBPatch* receiverPatch) {
    // get samples on receiver
    // - type==LT_W* ==> samples are ordered according to subpatches
    rSamples[] = receiverPatch->getSamples();
    // get samples on sender
    // - type==LT_W* ==> samples are ordered according to subpatches
    sSamples[] = senderPatch->getSamples();
}
```

³ Bei der Berechnung der Kernel-Werte werden die zugehörigen Sichtbarkeitswerte *nicht* mit einbezogen.

⁴ Kein Strahl wird blockiert.

```

// integrate
areaFraction = senderPatch->area / nSamples;
for (i=0;i<nSamples;i++) {
    // get indices of samples according to sampling-strategy
    (rIndex, sIndex) = getIndices(i);
    // visibility-in-quadrature
    determine visibility between rSamples[rIndex] and sSamples[sIndex]
    if (visible) {
        // geometric term *without* occlusion
        gTerm = geomTerm(rSamples[rIndex], receiverPatch->normal,
                        sSamples[sIndex], senderPatch->normal);
        kernel = gTerm * receiverPatch->brdf(rSamples[rIndex]);
        // simultaneous accumulation of kernel-values with respect
        // to the sign of the basisfunctions at the samples
        accumulateTCs(kernel * areaFraction, rIndex / nPosSamples,
                      sIndex / nPosSamples);
    }
}
}
}

```

3.1.3 Lösen der diskreten Radiance-Gleichungen

Zum Lösen der aktuellen diskreten Radiance-Gleichung wird das Gauss-Seidel-Verfahren eingesetzt. Die Berechnung des Skalarprodukts einer Zeile der Transportmatrix mit dem aktuellen Lösungsvektor erfolgt durch das *Einsammeln* von Radiance über die Links der zugehörigen Link-Liste. Alle Patch-Hierarchien und alle in den Patch-Hierarchien verwalteten Link-Listen werden vollständig traversiert. Der neue Koeffizient einer (Empfänger-)Basisfunktion wird als Summe von Produkten der in den Links verwalteten korrespondierenden Transportkoeffizienten mit den korrespondierenden Koeffizienten der Sender-Basisfunktionen berechnet. Bei der Berechnung des neuen Koeffizienten einer Mutter-Skalierungsfunktion wird als zusätzlicher Summand der vorgegebene korrespondierende Koeffizient der emittierenden Radiance-Verteilung berücksichtigt. Das Iterationsverfahren bricht ab, wenn sich die Radiance-Verteilung durch einen Iterationsschritt nur noch „geringfügig“ ändert – spätestens jedoch nach einer vorgegebenen maximalen Anzahl von Iterationsschritten. Für die Implementierung des Abbruchkriteriums wird während eines Iterationsschrittes die maximale Änderung einer Farbkomponente aller Radiance-Koeffizienten ermittelt. Falls dieser Wert unterhalb eines vorgegebenen Iterationsschwellwertes liegt, gilt das Verfahren als numerisch konvergiert. Das Verfahren konvergiert üblicherweise bereits nach ca. drei bis vier Iterationsschritten.

3.1.4 Verfeinerung der diskreten Radiance-Gleichungen

Ziel der Verfeinerung der aktuellen diskreten Radiance-Gleichung ist die Erweiterung der entsprechenden Transportmatrix um wenige signifikante Transportkoeffizienten mit Hilfe eines Orakels. Dabei werden ggf. auch Patch-Hierarchien dynamisch erweitert. In einem Verfeinerungsschritt werden die existierenden Transportkoeffizienten um höchstens eine Stufe entweder bzgl. der Empfänger-Basisfunktion oder bzgl. der Sender-Basisfunktion verfeinert. Die Verfeinerung orientiert sich an der Eltern-Kind-Beziehung. $T_{r \leftarrow s}$ wird bzgl. B_r verfeinert, indem die Transportmatrix für alle Kinder $B_{r'}$ von B_r um die Transportkoeffizienten $T_{r' \leftarrow s}$ erweitert wird. Dabei wird ggf. auch die Haar-Basis zur Darstellung der Radiance-Verteilung über dem Empfänger-Ausgangs-Patch um die $B_{r'}$ erweitert. Eine Verfeinerung bzgl. B_s erfolgt analog.

Aufgrund der verfolgten Linking-Strategie werden Transportkoeffizienten i. a. simultan verfeinert. Tabelle 3.2 dokumentiert die Anzahl und den Typ der feineren Links in Abhängigkeit des Typs des verfeinerten Links und dem „Verfeinerungsende“. In der Tabelle sind die Fälle mit X

	bzgl. des Empfängers		bzgl. des Senders
LT_SS	$1 \times \text{LT_WS}$		$1 \times \text{LT_SW}$
LT_WS	$4 \times \text{LT_WS}$	X	$3 \times \text{LT_WW}$
LT_SW	$3 \times \text{LT_WW}$		$4 \times \text{LT_SW}$
LT_WW	$4 \times \text{LT_WW}$	X	$4 \times \text{LT_WW}$

Tabelle 3.2 Verfeinerung von Links

markiert, in denen die feineren Links bzgl. des Empfängers auf einem feineren Level erzeugt werden. Beispielsweise erfolgt die Verfeinerung eines Links vom Typ `LT_WS` bzgl. des Empfängers durch das Erzeugen jeweils eines Links vom Typ `LT_WS` von den Kindern des Empfänger-Patches zu der Sender-Basisfunktion. Dabei werden ggf. die Enkel des Empfänger-Patches angelegt. Die Verfeinerung bzgl. des Senders erfolgt durch das Erzeugen jeweils eines Links vom Typ `LT_WW` vom Empfänger-Patch zu den drei Wavelets des Sender-Patches. Dabei werden ggf. die Kinder des Sender-Patches angelegt.

Links vom Typ `LT_WW` können auf zwei Arten entstehen: entweder durch die Verfeinerung eines Links bzgl. des Empfängers oder durch die Verfeinerung eines *anderen* Links bzgl. des Senders. Ein Link vom Typ `LT_WW` wird nur dann erzeugt, wenn die Link-Liste des Empfänger-Patches noch keinen derartigen Link enthält. Auf diese Weise werden Duplikate vermieden.

Kandidaten für die Verfeinerung sind alle Links, die noch *nicht* bzgl. beider Enden verfeinert sind. Jedem potentiell neuen Transportkoeffizienten wird durch das Orakel ein Signifikanzwert zugeordnet. Der Signifikanzwert eines Kandidaten ist definiert als das Maximum der Signifikanzwerte aller zu einer Verfeinerung des Kandidaten korrespondierenden potentiell neuen Transportkoeffizienten. In einem Verfeinerungsschritt werden nur diejenigen Kandidaten verfeinert, deren Signifikanzwerte einen bestimmten *Signifikanzschwellewert* übersteigen. Wenn ein derartiger Kandidat noch nicht verfeinert ist, wird von den beiden Verfeinerungsmöglichkeiten diejenige ausgewählt, die zu dem signifikantesten Transportkoeffizienten führt.

3.1.4.1 Orakel

Das Orakel schätzt die Signifikanz eines potentiell neuen Transportkoeffizienten $T_{r \leftarrow s}$ ab. Als Signifikanzmaß dient das Produkt der folgenden beiden skalaren Größen:

- Betrag der betragsmäßig maximalen Farbkomponente einer Abschätzung von $T_{r \leftarrow s}$
- Betrag der betragsmäßig maximalen Farbkomponente des Koeffizienten von B_s

$T_{r \leftarrow s}$ läßt sich bedingt durch den Einsatz der Haar-Basis gut und effizient durch die *Variation*⁵ der Kernel-Werte abschätzen, auf denen die Berechnung des potentiell *zu verfeinernden* Transportkoeffizienten basiert. Aus diesem Grund wird die Kernel-Variation bei der Berechnung der Transportkoeffizienten eines Links ermittelt und ihre maximale Farbkomponente in dem Link gespeichert. Intuitiv gilt ein potentiell neuer Transportkoeffizient als signifikant, wenn sein Absolutbetrag als „groß“ eingeschätzt wird *und* der Koeffizient der Sender-Basisfunktion „groß“ ist. Dieses Signifikanzmaß ist plausibel, da es gerade die Änderung des Koeffizienten der Empfänger-Basisfunktion abschätzt, falls die Transportmatrix um diesen Transportkoeffizienten erweitert würde. Das Orakel basiert im Vergleich zu dem im Rahmen des Hierarchical-Radiosity-Algorithmus üblicherweise verwendeten *BF-Orakels* [HSA91] auf zwei zusätzlichen Größen:

Sichtbarkeitsfunktion Die Kernel-Variation hängt von der Sichtbarkeitsfunktion ab, da die Sichtbarkeit bei der Berechnung der Transportkoeffizienten nach der Strategie *visibility-in-quadrature* berücksichtigt wird. Insbesondere werden Schattenübergänge gut aufgespiert. Wenn

⁵ Maximum - Minimum (komponentenweise)

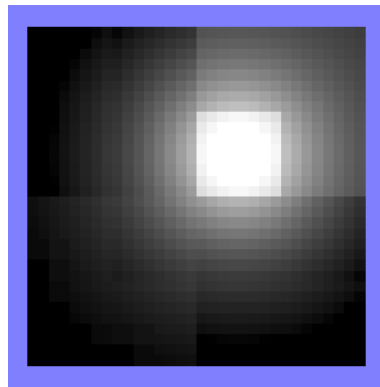
ein Strahl blockiert wird und ein weiterer Strahl nicht blockiert wird, ist der minimale Kernel-Wert 0 und die Kernel-Variation entsprechend „groß“.

BRDF Die Kernel-Variation hängt von der – im Radiosity-Fall für jeden Oberflächenpunkt konstanten – BRDF des Empfänger-Patches ab, da die BRDF Bestandteil des Kernels ist. Transportkoeffizienten zu schwach reflektierenden Empfängern gelten daher als unsignifikant.

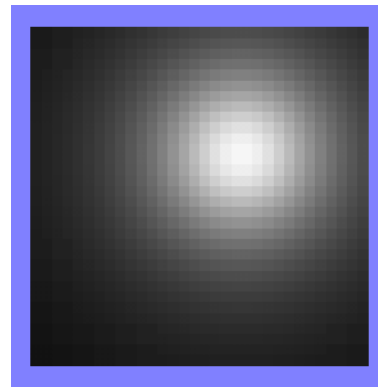
Der Signifikanzschwelligwert kann auf zwei Arten definiert werden: entweder *absolut* oder *relativ* zu dem Maximum der Signifikanzwerte aller Kandidaten. Bei der ersten Variante muß für jede Szene ein geeigneter Wert experimentell bestimmt werden. Demgegenüber zeichnet sich die zweite Variante durch ihren Automatismus aus. Die zweite Variante ist etwas aufwendiger, da das Maximum zunächst ermittelt werden muß. Falls man den Bruchteil sehr hoch ansetzt (z. B. 99%), wird die Transportmatrix automatisch in jedem Verfeinerungsschritt nur um sehr wenige Transportkoeffizienten erweitert.

3.1.4.2 Adaptive numerische Integration

Bei der verfolgten Standard-Zerlegung des Transportoperators werden verfeinerte Transportkoeffizienten *nicht* entfernt. Verfeinerte Transportkoeffizienten sind i. a. ihrerseits signifikant. Sie sollten daher möglichst genau berechnet werden. Die Grundidee besteht darin, mit Hilfe des bei der Berechnung der feineren Transportkoeffizienten verwendeten dichteren Samplings auch die Genauigkeit der verfeinerten Transportkoeffizienten und ggf. ihrer Vorfahren zu verbessern. Abbildung 3.2 veranschaulicht die Notwendigkeit der adaptiven numerischen Integration. Unge- nau berechnete grobe Transportkoeffizienten führen zu deutlich wahrnehmbaren Artefakten. Die Unterteilung des Ausgangsquadrats und die Unterteilung des rechten oberen Teilquadrats sind deutlich sichtbar.



(a) ohne adaptive numerische Integration



(b) mit adaptiver numerischer Integration

Abbildung 3.2 Direkte Beleuchtung eines Quadrats

Der Algorithmus für die adaptive numerische Integration basiert auf der two-scale-relationship. Jede Skalierungsfunktion und jedes Wavelet läßt sich als Linearkombination von um eine Auflösungsstufe feineren Skalierungsfunktionen darstellen. Folglich läßt sich auch ein Transportkoeffizient zwischen diesen Funktionen als Linearkombination von Transportkoeffizienten zwischen um eine Auflösungsstufe feineren Skalierungsfunktionen darstellen. Aus diesem Grund wird für jeden Link der Transportkoeffizient zwischen den beiden lokalen Skalierungsfunktionen mitverwaltet.

Nach einem kompletten Verfeinerungsschritt erfolgt die Verbesserung von „alten“ Transportkoeffizienten durch eine *bottom-up* Traversierung aller Links, so daß sich eine Verbesserung aus-

gehend von den „neuen“ Transportkoeffizienten bis eventuell zu den Transportkoeffizienten zwischen den Mutter-Skalierungsfunktionen fortsetzen kann. Im Rahmen dieser Arbeit wurde eine sehr einfache Variante der adaptiven numerischen Integration implementiert. Links vom Typ `LT_SS` werden ausschließlich durch Links vom Typ `LT_WS` verbessert⁶ und Links vom Typ `LT_WS`, `LT_SW` und `LT_WW` werden ausschließlich durch Links vom gleichen Typ verbessert.

Beispielsweise wird ein Link vom Typ `LT_WS` genau dann verbessert, wenn von jedem Kind des Empfänger-Patchs ein Link vom Typ `LT_WS` zu dem Sender-Patch existiert *und* wenn wenigstens einer dieser Links markiert ist. In einem Verfeinerungsschritt werden „neue“ Links und verbesserte Links markiert. Die Verbesserung basiert auf den Transportkoeffizienten zwischen den lokalen Skalierungsfunktionen, die in den Links von den Kindern des Empfänger-Patchs zu dem Sender-Patch verwaltet werden. Die verbesserten Transportkoeffizienten werden mit Hilfe der bereits bei der Berechnung der Transportkoeffizienten verwendeten Akkumulierungs-Prozedur berechnet. Für jede Kombination von Teil-Patches des Empfänger-Patchs mit dem Sender-Patch wird anstelle von mehreren Kernel-Werten der Transportkoeffizient zwischen den lokalen Skalierungsfunktionen akkumuliert.

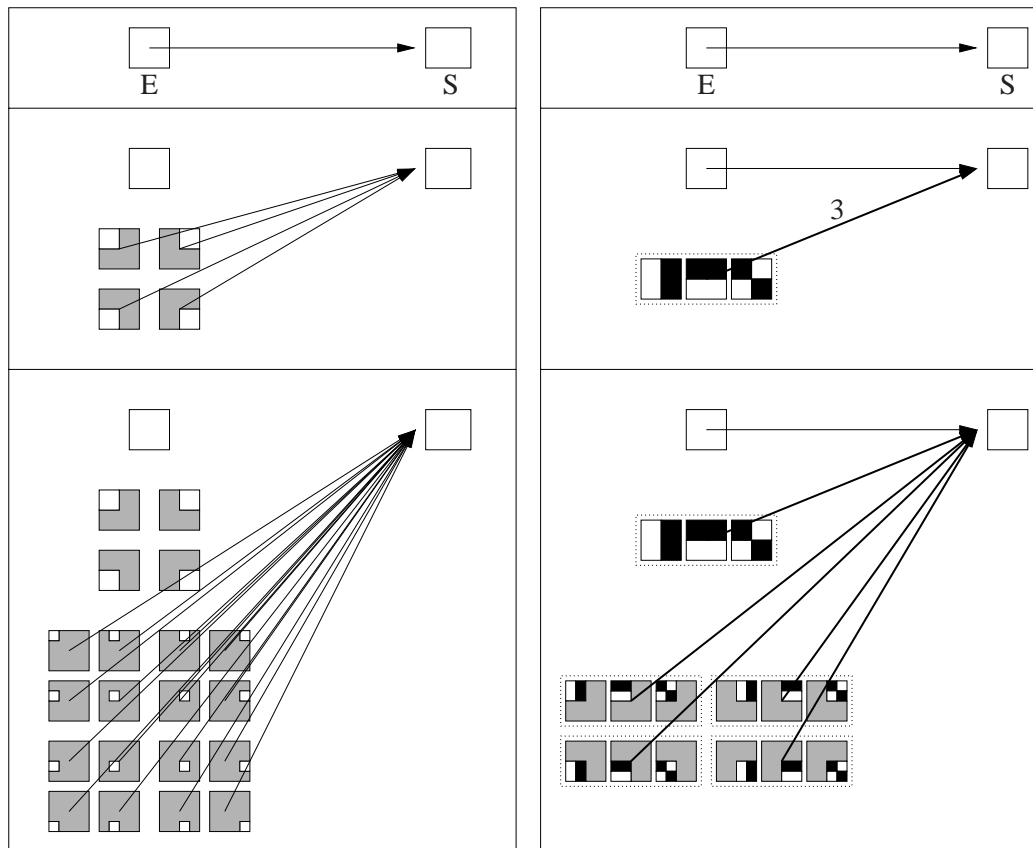
Eine ausgefeiltere Implementierung würde zusätzlich die Verbesserung von Links vom Typ `LT_SS` durch Links vom Typ `LT_SW` bzw. `LT_WW` in Betracht ziehen und die Verbesserung von Links vom Typ `LT_WS` bzw. `LT_SW` durch Links vom Typ `LT_WW`. Da ein Link in diesem Fall auf verschiedene Arten verbessert werden kann, müßte man ein Maß für die Güte von Transportkoeffizienten einführen, um die verschiedenen Möglichkeiten gegeneinander abwägen zu können. Da die implementierte einfache Variante bereits zu ausreichenden Verbesserungen führte, wurde auf die Entwicklung einer ausgefeilteren Variante verzichtet.

3.2 Vergleich mit dem Hierarchical-Radiosity-Algorithmus

Der klassische *Hierarchical-Radiosity-Algorithmus* [HSA91] ist ein Wavelet-basierter globaler Beleuchtungsalgorithmus, der auf der *Nicht-Standard-Zerlegung* des Transportoperators bzgl. der Haar-Basis basiert [SGCH93]. Außerdem kommt dieser Algorithmus unter Ausnutzung der *two-scale-relationship* ohne die explizite Verwaltung von Wavelet-Koeffizienten aus. In diesem Abschnitt sollen die wesentlichen Aspekte herausgestellt werden, in denen sich der Hierarchical-Radiosity-Algorithmus von dem auf der Standard-Zerlegung des Transportoperators basierenden Basis-Algorithmus unterscheidet.

Die Radiance-Verteilung wird über jedem Ausgangs-Patch bzgl. eines hierarchischen Erzeugendensystems bestehend aus Skalierungsfunktionen repräsentiert (vgl. linke Hälfte der Abbildung 2.10, Seite 26). Bei der Verfeinerung eines Transportkoeffizienten wird der verfeinerte Transportkoeffizient entfernt (vgl. Abbildung 3.3). Daher kommt der Hierarchical-Radiosity-Algorithmus ohne adaptive numerische Integration aus. Allerdings ist die Radiance-Verteilung über einem Ausgangs-Patch nach jedem Einsammeln von verschiedenen Sendern wegen der „Überrepräsentation“ i. a. in einem inkonsistenten Zustand. Durch eine vollständige Traversierung der Hierarchie kann die Radiance-Verteilung jedoch leicht in einen konsistenten Zustand gebracht werden. In der *top-down Phase (Push)* werden dabei die auf den verschiedenen Leveln ermittelten Radiance-Werte akkumuliert. Auf diese Weise wird jedem Blatt ein Radiance-Wert zugeordnet, der von allen Links abhängt, die entlang des Pfades zur Wurzel verwaltet werden. In der *bottom-up Phase (Pull)* werden die Radiance-Werte der Kinder gemittelt. Auf diese Weise wird jedem inneren Knoten ein Radiance-Wert zugeordnet, der von allen Links abhängt, die entlang der Pfade zu der Wurzel *und* zu allen korrespondierenden Blättern verwaltet werden.

⁶ Eine Verfeinerung bzgl. des Empfängers ist sehr wahrscheinlich.



(a) Hierarchical-Radiosity (Nicht-Standard-Zerl.)

(b) Wavelet-Radiosity (Standard-Zerlegung)

Abbildung 3.3 Fortschreitende (vollständige) Verfeinerung bzgl. des Empfängers

Die Implementierung des Hierarchical-Radiosity-Algorithmus ist im Vergleich zu der Implementierung des beschriebenen Wavelet-Radiosity-Algorithmus mit einem wesentlich geringeren Aufwand verbunden, da die Radiance-Koeffizienten und die Transportkoeffizienten in deutlich einfacheren Datenstrukturen verwaltet werden können. Pro Patch wird nur der korrespondierende Skalierungsfunktion-Koeffizient verwaltet und pro Link wird genau ein Transportkoeffizient zwischen zwei Skalierungsfunktionen verwaltet. Außerdem ist die Implementierung des Push-/Pull-Algorithmus im Vergleich zu der Implementierung der adaptiven numerischen Integration wesentlich unproblematischer.

3.3 Minimal Rendering Toolkit

3.3.1 Einführung

Das *Minimal Rendering Toolkit* (MRT) ist eine plattformunabhängige, objektorientierte und erweiterbare Bibliothek zur Visualisierung von dreidimensionalen Szenen [Fel96]. Die Bibliothek ist derzeit für die folgenden Betriebssysteme verfügbar: Irix (SGI), Solaris (SUN), Linux, Windows9X und WindowsNT. Die Bibliothek ist in der Programmiersprache C++ [Str91] geschrieben. Das MRT ist über viele Jahre hinweg unter der Regie von Prof. Dr. D. W. Fellner gewachsen. Primäre Einsatzgebiete sind Lehre und Forschung.

Das MRT ist eine Ansammlung von wohlstrukturierten Klassen, die für die Visualisierung von dreidimensionalen Szenen bedeutende abstrakte Konzepte kapseln. Wichtige abstrakte Konzepte sind z. B. geometrisches Objekt, Szene (hierarchisch strukturierte Menge von geometrischen Objekten), optische Oberflächeneigenschaften, Lichtquelle, Kamera, Bildebene und topologische Datenstruktur zur Verwaltung einer polygonalen Oberflächen-Approximation. Das Spektrum der angebotenen Visualisierungsarten reicht von dem einfachen, aber effizienten (ggf. hardwareunterstützten) Polygon-Rendering über strahlbasierte globale Beleuchtungsalgorithmen [Mül97] bis hin zu Hierarchical-Radiosity [BFS96].

Ein fundamentales Paradigma des MRT besteht darin, Objekt-Informationen solange wie möglich verfügbar zu halten. Bei einem approximativen Rendering wird jedes geometrische Objekt üblicherweise in einem frühen Stadium des Rendering-Prozesses durch eine polygonale Oberflächen-Approximation ersetzt. Auf Oberflächen-Approximationen arbeitende Algorithmen haben in diesem Fall keine Möglichkeit, auf die ursprünglichen geometrischen Objekte zuzugreifen. Demgegenüber verwaltet jedes geometrische MRT-Objekt bei Bedarf seine polygonale Oberflächen-Approximation, so daß derartige Algorithmen auf Objekt-Informationen zugreifen können.

Das Ausnutzen von Objekt-Informationen hat bei der bereits existierenden Implementierung des Hierarchical-Radiosity-Algorithmus zu einer Reihe von Verbesserungen sowohl hinsichtlich der Genauigkeit als auch hinsichtlich der Effizienz geführt. Einige Verbesserungen ließen sich direkt bei der Implementierung des Wavelet-Radiosity-Algorithmus bzw. des Wavelet-Radiance-Algorithmus übernehmen. Im Abschnitt 3.4.2.3 wird gezeigt, wie sich auch beim Final-Gathering die visuelle Qualität der Darstellung von gekrümmten Objekten unter Ausnutzung von Objekt-Informationen verbessern läßt.

3.3.2 Anbindung des Basis-Algorithmus

Die Anbindung des Basis-Algorithmus an das MRT orientiert sich eng an der Anbindung des Hierarchical-Radiosity-Algorithmus. Die zu den Blättern der Patch-Hierarchien eines geometrischen Objekts korrespondierenden Dreiecke bzw. planaren Vierecke werden in einer topologischen Datenstruktur verwaltet [Ben97]. Hauptkomponenten dieser Datenstruktur sind Flächen, Halbkanten und Knoten. Neben der Kompaktheit und der Konsistenz zeichnet sich diese Datenstruktur vor allem durch die effiziente Verfügbarkeit von Nachbarschafts-Informationen aus. Die Implementierung des Basis-Algorithmus profitiert insbesondere hinsichtlich der folgenden Aspekte von der Anbindung an das MRT:

Sichtbarkeitsberechnungen Die Sichtbarkeitsberechnungen basieren auf den ausgefeilten *Strahlanfrage*⁷-Methoden des MRT. Die Berechnung der Sichtbarkeit zwischen zwei auf verschiedenen Patches platzierten Samples erfolgt durch eine an die Szene gerichtete Strahlanfrage. Die potentiell blockierenden Objekte werden dabei nicht durch ihre mit fortschreitender Verfeinerung i. a. immer komplexer werdende polygonale Approximation repräsentiert, sondern – unter Ausnutzung von Objekt-Informationen – kompakt durch ihre ursprüngliche Definition. Die unterschiedlichen von dem MRT zur Verfügung gestellten Verfahren zur Beschleunigung von Strahlanfragen (Hüllkörperhierarchien, Raumunterteilungen, Hybridverfahren) können auch für die Beschleunigung der Sichtbarkeitsberechnungen eingesetzt werden. Da der Zeitaufwand für die Sichtbarkeitsberechnungen den Zeitaufwand für die gesamte Radiance-Berechnung dominiert, wirkt sich eine Effizienz-Verbesserung in diesem speziellen Bereich deutlich auf die gesamte Laufzeit des Algorithmus aus.

⁷ Feststellung des relativ zum Strahlursprung innerhalb eines vorgegebenen Intervalls nächstgelegenen Schnittpunktes mit der Szene [Mül97]

Rekonstruktion durch Gouraud-Shading Die bzgl. einer stückweise konstanten Basis repräsentierte Radiance-Verteilung weist an den Patch-Grenzen i. a. visuell störende Helligkeitssprünge auf (vgl. Abbildung 3.2(b), Seite 38). Bei der graphischen Ausgabe ist eine „Glättung“ der berechneten Radiance-Verteilung durch eine *Rekonstruktionsmethode* erforderlich. Eine wegen ihrer Effizienz sehr populäre Methode ist das *Gouraud-Shading*. Das hardwareunterstützte Gouraud-Shading ermöglicht die *interaktive* graphische Ausgabe von Lösungen für Szenen mittlerer Komplexität.

Bei der Initialisierung des Gouraud-Shadings wird jedem Paar (Knoten, angrenzende Fläche) ein Farbwert zugeordnet. Dieser Farbwert ist im Fall von nicht-gekrümmten Objekten proportional zu dem Mittelwert der Radiance-Werte derjenigen Flächen, die an den Knoten grenzen *und* dieselbe Orientierung wie die Bezugs-Fläche aufweisen. Das Gouraud-Shading selbst korrespondiert zu der linearen Interpolation dieser Farbwerte über der Fläche. Die zweite Bedingung stellt dabei sicher, daß nicht über Begrenzungskanten hinweg interpoliert wird. Die Verwaltung von Nachbarschafts-Informationen ist entscheidend für eine effiziente Initialisierung des Gouraud-Shadings.

Tabelle 3.3 dokumentiert das die Anbindung realisierende Ableitungs-Schema. Die Basis-Klassen auf der linken Seite sind Bestandteile des MRT. Die Klasse `t_IllumScene` kapselt das Konzept einer beleuchteten Szene. Sie stellt die virtuellen Methoden `radiosityInit()` und `radiosityStep()` zur Verfügung. Diese Methoden werden in der abgeleiteten Klasse überschrieben. Die Initialisierungs-Methode sendet zunächst die Nachricht `approxShape()` an die Szene, wodurch jedes geometrische Objekt seine polygonale Approximation erzeugt. Danach wird für jede erzeugte Fläche eine Patch-Hierarchie angelegt, die anfangs nur aus dem Ausgangs-Patch besteht. Zwischen allen gegenseitig zugewandten und wenigstens teilweise sichtbaren geordneten Patch-Paaren wird ein Link vom Typ `LT_SS` eingerichtet (initial linking). Die Schritt-Methode führt alternierend einen Lösungs- bzw. einen Verfeinerungs-Schritt durch. Jedes geometrische Objekt verweist auf eine Instanz der Klasse `t_SurfaceShader`, die die optischen Oberflächeneigenschaften verwaltet. Die Klasse `t_RadPatch` kapselt das Konzept des Elements einer Patch-Hierarchie. Die Ableitung stellt insbesondere die Verbindung zur topologischen Datenstruktur her. Auf diese Weise kann auch die Lösung einer Wavelet-Radiosity- bzw. Wavelet-Radiance-Berechnung effizient durch Gouraud-Shading geglättet graphisch ausgegeben werden. Die Basis-Klasse wurde um Methoden zur Bereitstellung von regulär bzw. gleichverteilt zufällig platzierten Samples erweitert, so daß diese Methoden sowohl im Radiosity-Fall als auch im Radiance-Fall zur Verfügung stehen. Die Klasse `t_DLink` kapselt das Konzept des Elements einer verketteten Liste. Die vier Link-Typen sind durch Ableiten von einer abstrakten Basis-Klasse implementiert.

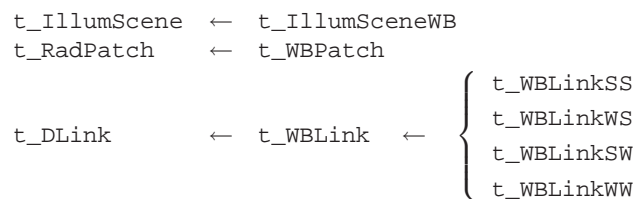


Tabelle 3.3 Ableitungs-Schema

3.4 Erweiterungen

3.4.1 Gekrümmte Objekte

Das Hauptaugenmerk bei der Implementierung des Hierarchical-Radiosity-Algorithmus lag auf der Integration von gekrümmten geometrischen Objekten [Sch97]. Im folgenden werden die Problematik und der verfolgte Lösungsansatz skizziert. Auf analoge Weise erfolgte auch die Integration von gekrümmten Objekten in den Wavelet-Radiosity-Algorithmus. Die Oberfläche eines gekrümmten Objekts läßt sich durch ein polygonales Netz nur – beliebig genau – *approximieren*. Eine z. B. für die graphische Ausgabe erforderliche gute Approximation umfaßt viele kleine Begrenzungsflächen. Ein trivialer Ansatz besteht darin, gekrümmte Objekte a priori fein zu unterteilen. Dieser Ansatz paßt jedoch nicht zu der Philosophie von Wavelet-basierten globalen Beleuchtungsalgorithmen, die Verfeinerung automatisch in Abhängigkeit eines Signifikanzmaßes durchzuführen. Da der Zeit- und Speicheraufwand von Wavelet-basierten globalen Beleuchtungsalgorithmen quadratisch mit der Anzahl der Ausgangs-Patches zunimmt, ist dieser Ansatz zu teuer. Statt dessen werden gekrümmte Objekte a priori möglichst grob unterteilt. Bei der Verfeinerung werden neu erzeugte Knoten der topologischen Datenstruktur aus der Flächen-Ebene – unter Ausnutzung von Objekt-Informationen – auf die exakte Oberfläche verschoben. Auf diese Weise verbessert sich mit zunehmender Ausführungszeit des Algorithmus neben der Radiance-Verteilung auch die geometrische Approximation. Die geometrische Approximation in schwach beleuchteten Bereichen wird ggf. a posteriori verbessert. Abschließend sollen die beiden in dem Abschnitt 3.3.2 beschriebenen Aspekte hinsichtlich der Integration von gekrümmten Objekten noch einmal aufgegriffen werden:

Sichtbarkeitsberechnungen Die Darstellung der potentiell blockierenden Objekte durch ihre ursprüngliche exakte Definition führt neben der beschriebenen Effizienz-Verbesserung auch zu einer Genauigkeits-Verbesserung, da gekrümmte Objekte anstelle eines „approximativen Schattens“ einen „exakten Schatten“ werfen.

Rekonstruktion durch Gouraud-Shading Der bei der Initialisierung des Gouraud-Shadings einem Paar (Knoten, angrenzende Fläche) zugeordnete Farbwert ist im Fall von gekrümmten Objekten proportional zu der gewichteten Summe der Radiance-Werte der an den Knoten grenzenden Flächen. Auf diese Weise wird über „virtuelle“ Begrenzungskanten hinweg interpoliert.

3.4.2 Final-Gathering

Im Abschnitt 3.3.2 wurde bereits der Einsatz einer Rekonstruktionsmethode motiviert und mit dem Gouraud-Shading eine sehr effiziente Methode beschrieben. In diesem Abschnitt wird eine alternative, strahlbasierte Rekonstruktionsmethode vorgestellt [Rei92]. Zunächst wird der Basis-Final-Gathering-Algorithmus – einschließlich seiner Implementierung – beschrieben. Danach werden Beschleunigungs-Techniken aufgezeigt. Schließlich wird die Integration von gekrümmten Objekten erläutert.

3.4.2.1 Basis-Algorithmus

Das Final-Gathering gleicht bis zu dem ersten von einem Sehstrahl getroffenen Oberflächenpunkt dem klassischen Raytracing. Dann erfolgt für jeden zu einem Pixel-Mittelpunkt korrespondierenden Oberflächenpunkt ein letztes Einsammeln von Radiance aus der Umgebung basierend auf der berechneten approximativen Lösung. Im Vergleich zu dem Gouraud-Shading führt das Final-Gathering zu Bildern höherer visueller Qualität. Ein zu einem Blatt-Patch korrespondierender Scanline-Abschnitt kann einen nicht-linearen Intensitätsverlauf annehmen. Insbesondere

lassen sich Schattenübergänge sehr realitätsnah visualisieren. Außerdem können *surface-detail*-Techniken⁸ aus dem Raytracing-Bereich zur weiteren Erhöhung der visuellen Qualität eingesetzt werden. Das Final-Gathering ist jedoch wesentlich zeitaufwendiger als das Gouraud-Shading, da für jedes von einem geometrischen Objekt überdeckte Pixel ein Einsammeln von Radiance aus der gesamten Umgebung (inkl. Sichtbarkeitsberechnungen) durchgeführt werden muß. Ein mögliches Einsatzgebiet ist z. B. das Erstellen von Referenzlösungen.

Das Final-Gathering wird erst in Verbindung mit Wavelet-basierten globalen Beleuchtungsalgorithmen einigermaßen praktikabel. In diesem Fall kann die Radiance aus der Umgebung gemäß einem geeigneten *level-of-detail* berücksichtigt werden, indem sie über die Link-Datenstruktur eingesammelt wird, die während der Berechnung der approximativen Lösung aufgebaut wurde [LTG93]. Im folgenden werden die wesentlichen Implementierungs-Details genauer beschrieben. Gesucht ist die von einer vorgegebenen Differential-Fläche dx auf der Oberfläche eines geometrischen Objekts in die Sichtrichtung $\vec{\omega}$ zum Augpunkt ausstrahlende Radiance.

Ermittlung eines Blatt-Patches

Das Ausnutzen der Link-Datenstruktur setzt eine Methode voraus, die zu x ein korrespondierendes Blatt der Patch-Hierarchie ermittelt. Im Fall eines nicht-gekrümmten Objekts ist diese Zuordnung unproblematisch, da alle Oberflächenpunkte exakt auf der polygonalen Approximation liegen. Die Zuordnung kann in diesem Fall sehr effizient durch das Ausnutzen der während der Berechnung der approximativen Lösung zu den Ausgangs-Patches aufgebauten Patch-Hierarchien vorgenommen werden. Zunächst wird der Ausgangs-Patch ermittelt, auf dem x liegt. Anschließend wird die mit dem Ausgangs-Patch assoziierte Patch-Hierarchie als Suchbaum verwendet. Der auf diese Weise ermittelte Blatt-Patch dient als „Eintrittspunkt“ in die Link-Datenstruktur.

Einsammeln von Radiance über die Link-Datenstruktur

Das Final-Gathering wurde zunächst für den Hierarchical-Radiosity-Algorithmus implementiert. In diesem Fall kann sich das Einsammeln von Radiance aus der Umgebung *direkt* an der Link-Datenstruktur orientieren. Für alle Patches entlang des Pfades von dem Blatt-Patch zu der Wurzel werden die korrespondierenden Listen von Links traversiert. Der Koeffizient der zu einem derartigen Link korrespondierenden Sender-Basisfunktion trägt den folgenden Anteil zu der gesuchten Radiance bei:

$$T_{x \leftarrow s} L_s \quad (3.1)$$

Bei dem Punkt-Basisfunktion-Transportkoeffizienten $T_{x \leftarrow s}$ handelt es sich um eine vereinfachte Variante des gewöhnlichen Transportkoeffizienten zwischen zwei Basisfunktionen. Die Integration über den Träger einer Empfänger-Basisfunktion entfällt. Statt dessen bezieht sich der Transportkoeffizient ausschließlich auf die Differential-Fläche dx :

$$T_{x \leftarrow s} \equiv f_r(x) \int_{x' \in \text{supp}(B_s)} g(x, x') B_s(x') dx'$$

Bei der Implementierung des Final-Gatherings für den Wavelet-Radiosity-Algorithmus könnte sich das Einsammeln von Radiance ebenfalls direkt an der Link-Datenstruktur orientieren. Das Hauptproblem bei diesem Ansatz wäre die genaue Berechnung von Transportkoeffizienten bzgl. grober (Sender-)Basisfunktionen. Man müßte eine vereinfachte Variante der adaptiven numerischen Integration implementieren. Ferner müßte man darauf achten, daß von einer (Sender-)Basisfunktion nur einmal Radiance eingesammelt wird, da mehrere Links entlang des Pfades zu der Wurzel auf dieselbe Basisfunktion zeigen können (vgl. Abbildung 3.3(b), Seite 40).

⁸ z. B. texture-mapping oder bump-mapping

Statt dessen wurde ein alternativer Ansatz entwickelt und implementiert, bei dem sich das Einsammeln von Radiance *indirekt* an der Link-Datenstruktur orientiert. Die Grundidee besteht darin, die beim Final-Gathering für den Hierarchical-Radiosity-Algorithmus auftretende Situation zu *simulieren*. Das Einsammeln von Radiance über einem Sender-Ausgangs-Patch erfolgt nicht anhand der Darstellung der Radiance-Verteilung bzgl. der Nicht-Standard-Haar-Basis, sondern bzgl. der korrespondierenden kanonischen Basis. Voraussetzung für diesen Ansatz ist die Verwaltung der Skalierungsfunktion-Koeffizienten für *jeden* Patch – nicht nur für Wurzeln und Blätter. Die Berechnung dieser Koeffizienten erfordert keinen zeitlichen Mehraufwand, da die Koeffizienten als Zwischenergebnisse bei der Ausführung des Pyramiden-Algorithmus anfallen. Nach der Ausführung des Pyramiden-Algorithmus ist die Radiance-Verteilung über einem Ausgangs-Patch zusätzlich bzgl. eines hierarchischen Erzeugendensystems bestehend aus Skalierungsfunktionen – analog zum Hierarchical-Radiosity-Algorithmus – repräsentiert. Für jede mit x assoziierte Sender-Hierarchie wird diejenige Teil-Hierarchie bestimmt, die gemäß der Link-Datenstruktur signifikant zur Beleuchtung von x beiträgt und somit den geeigneten *level-of-detail* festlegt. Dazu werden die zu den Patches entlang des Pfades von dem Blatt-Patch zu der Wurzel korrespondierenden Listen von Links traversiert, und alle Patches markiert, auf die ein Link vom Typ `LT_*W` verweist. Das Einsammeln von Radiance erfolgt schließlich gemäß der Formel 3.1 bzgl. derjenigen lokalen Skalierungsfunktionen, die unmittelbar unterhalb der untersten Stufe der markierten Teil-Hierarchien verwaltet werden⁹.

Berechnung der Transportkoeffizienten

Die Punkt-Patch-Sichtbarkeit wird *nicht* nach der Strategie *visibility-in-quadrature* berücksichtigt, sondern gemäß einem eigenen Satz von Strahlen ermittelt. Diese Entkoppelung ist für die Implementierung einiger Beschleunigungs-Techniken vorteilhaft. Der unverdeckte Punkt-Skalierungsfunktion-Transportkoeffizient wird exakt mit Hilfe einer Formel zur analytischen Berechnung des Punkt-Patch-Formfaktors [Gla95] berechnet.

Anbindung an das MRT

Die Klasse `t_IllumScene` stellt die virtuelle Methode `incidentLight()` zur Verfügung, die im Raytracing-Kontext zu einem vorgegebenen Sehstrahl die aus der Strahlrichtung einfallende Licht-Energie berechnet. Die Anbindung des Final-Gatherings an das MRT erfolgt durch das Überschreiben dieser Methode in der abgeleiteten Klasse `t_IllumSceneWB`. Zu dem vorgegebenen Sehstrahl wird eine Strahlanfrage an die Szene gerichtet. Falls ein Objekt getroffen wird, wird ein Blatt-Patch ermittelt, und die Radiance aus der Umgebung über die Link-Datenstruktur eingesammelt. Das Final-Gathering wird durch das Senden der Nachricht `rayTrace()` an eine Instanz der Klasse `t_Image` angestoßen. Die existierende Methode `t_Image::rayTrace()` sendet für jeden Sehstrahl eine Nachricht `incidentLight()` an die Szenen-Instanz. Auf diese Weise werden bei der Implementierung des Final-Gatherings die konzeptionellen Gemeinsamkeiten mit dem Raytracing ausgenutzt. Das Raytracing wird gewissermaßen durch das Final-Gathering „überschrieben“. Darüber hinaus profitiert die Implementierung des Final-Gatherings ebenfalls insbesondere hinsichtlich der Sichtbarkeitsberechnungen von der Anbindung an das MRT (vgl. Abschnitt 3.3.2).

⁹ Falls kein Patch einer mit x assoziierten Sender-Hierarchie markiert wurde (da nur Links vom Typ `LT_*S` auf diese verweisen), erfolgt das Einsammeln von Radiance bzgl. der Mutter-Skalierungsfunktion.

3.4.2.2 Beschleunigungs-Techniken

Das Final-Gathering erfordert keinen zusätzlichen Speicherplatz, da die Punkt-Skalierungsfunktion-Transportkoeffizienten im Gegensatz zu den gewöhnlichen Transportkoeffizienten nicht gespeichert werden brauchen. Allerdings ist das Final-Gathering sehr zeitaufwendig. Der Schlüssel für eine Reduktion des Zeitaufwands ist auch in diesem Fall eine effizientere Sichtbarkeitsberechnung. Im folgenden werden einige Techniken beschrieben, die zur Beschleunigung der Sichtbarkeitsberechnungen eingesetzt worden sind. In dem Abschnitt 3.5 wird der für eine Beispiel-Szene ermittelte Zeitgewinn dokumentiert.

A Getrennte Behandlung von direkter und indirekter Beleuchtung Diese Strategie ist durch die folgenden Beobachtungen motiviert:

- I. a. entfällt nur ein geringer Prozentsatz der Transporte auf direkte Beleuchtung.
- Durch direkte Beleuchtung hervorgerufene Schattenübergänge müssen sehr genau dargestellt werden, um eine hohe visuelle Qualität zu erreichen.
- Die Genauigkeit der Darstellung von Schattenübergängen nimmt mit der Anzahl der Schattenstrahlen zu. Bei Verwendung von nur einem Schattenstrahl werden alle Schattenübergänge abrupt dargestellt. Je mehr Schattenstrahlen verwendet werden, desto genauer können von flächigen Lichtquellen verursachte weiche Schattenübergänge visualisiert werden.

Die Anzahl der Schattenstrahlen pro Transport von indirekter Beleuchtung kann i. a. deutlich geringer als die Anzahl der Schattenstrahlen pro Transport von direkter Beleuchtung gewählt werden. Auf diese Weise läßt sich der Zeitaufwand – ohne wahrnehmbaren Qualitätsverlust – deutlich reduzieren.

B Adaptive Anzahl der Schattenstrahlen für indirekte Beleuchtung Die Anzahl der Schattenstrahlen pro Transport von indirekter Beleuchtung wird proportional zu dem unverdeckten Punkt-Patch-Formfaktor gewählt. Der unverdeckte Punkt-Patch-Formfaktor ist definiert als der Anteil der Basisfläche der Einheitshalbkugel, der von der Orthogonalprojektion des korrespondierenden Raumwinkels auf die Basisfläche eingenommen wird. Auf diese Weise paßt sich die Anzahl automatisch an die Größe des Sender-Patches und die relative Lage von dx zu dem Sender-Patch an. Je kleiner der Sender-Patch, je größer die Transportentfernung, je schiefer die Lichteinstrahlung, desto weniger Schattenstrahlen werden verwendet.

C Ausnutzen der Patch-Patch-Sichtbarkeit Diese Strategie basiert auf der folgenden trivialen Beobachtung: Wenn ein Patch-Paar gegenseitig vollständig sichtbar bzw. verdeckt ist, gilt diese Sichtbarkeitsbeziehung insbesondere für jeden Punkt eines der beiden Patches gegenüber dem anderen Patch. Daher verwaltet jeder Link zusätzlich die Sichtbarkeit¹⁰ zwischen seinem Empfänger-Patch und seinem Sender-Patch. Diese Patch-Patch-Sichtbarkeit fällt bei der Berechnung der gewöhnlichen Transportkoeffizienten als Zwischenergebnis an. Bei der Berechnung eines Punkt-Skalierungsfunktion-Transportkoeffizienten wird die Punkt-Patch-Sichtbarkeit nur dann mittels Strahlschuß berechnet, wenn die korrespondierenden Patches gegenseitig teilweise sichtbar sind. Vollständige Sichtbarkeit bzw. Verdeckung wird übernommen. Diese Strategie kann jedoch – ähnlich der Vererbung von vollständiger Sichtbarkeit bei der Berechnung von gewöhnlichen Transportkoeffizienten – zu Artefakten führen, falls ein Patch-Paar z. B. bedingt durch ein sehr grobes Sampling *fälschlicherweise* als gegenseitig vollständig sichtbar bzw. verdeckt eingestuft wird. Diese Fehleinschätzung könnte durch die Berechnung der Punkt-Patch-Sichtbarkeit revidiert werden. Aus diesem Grund

¹⁰ diskreter Wertebereich: {vollständig sichtbar, teilweise sichtbar, vollständig verdeckt}

läßt sich diese Optimierung – getrennt für direkte und indirekte Beleuchtung – bei Bedarf deaktivieren. Da sich das Einsammeln von Radiance nicht direkt an der Link-Datenstruktur orientiert, ist der Zugriff auf die korrespondierende Patch-Patch-Sichtbarkeit nicht trivial. Im Rahmen der Markierung der Teil-Hierarchien wird die von einem Link verwaltete Patch-Patch-Sichtbarkeit dem korrespondierenden Sender-Patch genau dann zugeordnet, wenn diesem noch keine Sichtbarkeit zugeordnet worden ist. Die letzte Bedingung stellt zusammen mit der bottom-up Vorgehensweise bei der Markierung sicher, daß jedem Patch die genaueste verfügbare Sichtbarkeitsinformation zugeordnet wird. Ferner erfolgt der Zugriff auf die Patch-Patch-Sichtbarkeit über den Vater-Patch.

D Ausnutzen von Strahlkohärenz durch einen Blockierer-Cache Bei der Ermittlung der Punkt-Patch-Sichtbarkeit durch einen Satz von Strahlen wird das zuletzt aufgespürte blockierende Objekt zwischengespeichert. Eine Strahlanfrage wird zunächst an den letzten Blockierer gerichtet. Die Anfrage wird nur dann an die gesamte Szene weitergeleitet, falls dabei kein Schnittpunkt festgestellt wird. Aufgrund der Strahlkohärenz ist die Wahrscheinlichkeit hoch, daß alle blockierten Strahlen eines Satzes von demselben Objekt blockiert werden. In diesem Fall braucht das blockierende Objekt nur einmal ermittelt zu werden – unabhängig von der Genauigkeit der Darstellung der Schattenübergänge.

E Reguläre Raumenterteilung Diese von dem MRT zur Verfügung gestellte Technik zur Beschleunigung von Strahlanfragen ist in [Mül97] dokumentiert.

3.4.2.3 Gekrümmte Objekte

In diesem Abschnitt wird gezeigt, wie die visuelle Qualität einer Final-Gathering-Darstellung unter Ausnutzung von Objekt-Informationen um eine weitere Stufe erhöht werden kann: die geometrisch exakte¹¹ Darstellung von gekrümmten Objekten. Falls die Objekt-Informationen in einem frühen Stadium des Rendering-Prozesses verloren gehen, können gekrümmte Objekte nur approximativ gemäß ihrem polygonalen Netz ausgegeben werden. Das MRT-Design ermöglicht hingegen eine – analog zum Raytracing – geometrisch exakte Darstellung von gekrümmten Objekten. Die Radiance aus der Umgebung wird auch für gekrümmte Objekte bzgl. Differential-Flächen eingesammelt, die exakt auf der Oberfläche liegen.

Eine Strahlanfrage an die Szene zu einem vorgegebenen Sehstrahl liefert im Treffer-Fall die benötigten Objekt-Informationen zurück:

- einen Zeiger auf das getroffene geometrische Objekt
- die Koordinaten des getroffenen Oberflächenpunktes x
- die Koordinaten der Oberflächennormalen n_x an der Stelle x

Das Hauptproblem besteht darin, den geeigneten „Eintrittspunkt“ in die Link-Datenstruktur zu finden. Die Ermittlung des zu x korrespondierenden Blatt-Patches der Patch-Hierarchie ist nicht trivial, da x im Fall eines gekrümmten Objekts i. a. nicht exakt auf der polygonalen Approximation liegt. Intuitiv sollte x ein Blatt-Patch zugeordnet werden, dessen Abstand zu x gering ist und dessen Normale nur geringfügig von n_x abweicht. Dieses Problem ist gewissermaßen dual zu dem Problem, einen Knoten der polygonalen Approximation auf die Oberfläche zu verschieben. Der folgende zweistufige Lösungsansatz hat sich in der Praxis bewährt. Grundsätzlich wird die Zuordnung über eine Strahlanfrage *an die polygonale Approximation* des Objekts vorgenommen. Aus der Menge aller Begrenzungspolygone wird ein Polygon mit minimalem Abstand von x entlang $-n_x$ ausgewählt. Wenn die Strahlanfrage zu keinem Treffer führt¹²,

¹¹ im Rahmen der Auflösung des Ausgabemediums

¹² Bsp.: Kreisscheibensegment eines Zylinders, das durch die polygonale Approximation nicht abgedeckt wird

wird ein Polygon mit minimalem Abstand von x ausgewählt. Dabei werden Mehrdeutigkeiten durch Vergleichen der Abweichung der Patch-Normalen von n_x aufgelöst. Die Berechnung eines Punkt-Skalierungsfunktion-Transportkoeffizienten basiert ferner auf der exakten Oberflächennormalen n_x . Der geometrische Term hängt von der Normalen ab.

3.5 Ergebnisse

In diesem Abschnitt werden die Ergebnisse von einigen Wavelet-Radiosity-Berechnungen dokumentiert. Das Hauptaugenmerk liegt auf einem visuellen Vergleich der beiden beschriebenen Rekonstruktionsmethoden und auf einer Untersuchung des Zeitgewinns, der bei dem Final-Gathering durch die beschriebenen Beschleunigungs-Techniken erzielt werden kann. Die dokumentierten Zeitspannen wurden bei der Ausführung des Algorithmus auf einer SGI Indigo 2 (R4400/200 MHz) gemessen. Die als Eingabe dienende Szene besteht aus einem „offenen“ Raum, in dem sich u. a. ein konkaver Superquadric, ein Zylinder und eine Anordnung von Kugeln befinden (vgl. Abbildung 3.5). Die Szene wird hauptsächlich von einer Lichtquelle direkt beleuchtet. Eine zweite Lichtquelle dient zur direkten Beleuchtung der Weltkarte.

Die Abbildung 3.6 zeigt die nach sechs Verfeinerungsschritten berechnete approximative Lösung des globalen Beleuchtungsproblems *ohne* den Einsatz einer Rekonstruktionsmethode. Der „Kachel-Effekt“ bedingt durch die Verwendung einer stückweise konstanten Basis für die Darstellung der Radiance-Verteilung ist deutlich sichtbar. Der Zeitaufwand für die Sichtbarkeitsberechnungen beträgt 62 Sekunden. Die Abbildung 3.7 stellt das der Lösung zugrunde liegende polygonale Netz dar. Sie verdeutlicht das gute Aufspüren von Schattenübergängen.

Die Abbildungen 3.4 und 3.5 zeigen die durch die beschriebenen Rekonstruktionsmethoden geglätteten Lösungen. Der Einfluß von ausschließlich direkter bzw. indirekter Beleuchtung auf die Final-Gathering-Darstellung ist in den Abbildungen 3.8 und 3.9 dokumentiert. Dabei wurde das Ergebnis der indirekten Beleuchtung um den Faktor 5 aufgehellt. Das Final-Gathering führt im Vergleich zu dem Gouraud-Shading zu einem deutlich brillanteren Bild. Bemerkenswert sind insbesondere die sehr realitätsnahen weichen Schattenübergänge. Bei dem Final-Gathering wird sogar der schmale Schattenbereich entdeckt, den die untere Spitze des Superquadrics auf die Oberseite des Zylinders wirft. Ferner werden auch solche für Radiosity-Lösungen charakteristischen *color-bleeding*-Effekte aufgespürt, die aufgrund eines hohen Signifikanzschwellwertes oder einer geringen Anzahl von Verfeinerungsschritten in der berechneten Lösung fehlen. Color-bleeding bezeichnet das „Ausbluten“ des Farbtons eines geometrischen Objekts in seine Umgebung aufgrund von diffuser Reflexion (vgl. Abbildung 3.9). Die stärksten color-bleeding-Effekte sind auch in der Abbildung 3.5 wahrnehmbar, sofern sie nicht von direkter Beleuchtung überlagert werden. Beispielsweise erfährt die Unterseite der weißen Kugel eine leichte Grün-Färbung und die Oberseite des Zylinders eine leichte Rot-Färbung. Die Abbildungen 3.10 und 3.11 verdeutlichen anhand eines Szenen-Details den Qualitätsgewinn, der durch die in dieser Arbeit entwickelte Integration von gekrümmten Objekten erzielt werden konnte. Bei dem Gouraud-Shading werden gekrümmte Objekte approximativ dargestellt, bei dem Final-Gathering hingegen exakt.

Die Untersuchung der quantitativen Auswirkungen der Beschleunigungs-Techniken für das Final-Gathering (vgl. Abschnitt 3.4.2.2) basiert auf einer approximativen Lösung nach drei Verfeinerungsschritten. Bei einer höheren Anzahl verursacht die Technik C kleinere Artefakte hinsichtlich direkter Beleuchtung. Die dokumentierten absoluten Meßergebnisse beziehen sich auf eine Auflösung von 200^2 Pixeln. Die absoluten Ergebnisse sind proportional zu der gewählten Auflösung und die relativen Ergebnisse sind somit invariant gegenüber der Auflösung. Nur rund 1% der ca. 4707000 Transporte entfallen auf direkte Beleuchtung. Pro Transport von direkter Beleuchtung werden 128 Schattenstrahlen verwendet, um den zur weißen Kugel korrespondierenden weichen Schatten möglichst realitätsnah darzustellen. Pro Transport von indirekter Be-

leuchtung werden *adaptiv* 192 – de facto im Durchschnitt nur rund 0.6 – Schattenstrahlen verwendet. Die Beschleunigungs-Techniken **A** und **B** ermöglichen die Berechnung einer Lösung von hoher visueller Qualität mit insgesamt im Durchschnitt nur rund 1.9 Schattenstrahlen pro Transport. 68% der Schattenstrahlen entfallen auf direkte Beleuchtung. Der Zeitaufwand für die Sichtbarkeitsberechnungen beträgt insgesamt 654s. Der Zeitaufwand für die Sichtbarkeitsberechnungen beim Einsammeln von direkter Beleuchtung (451s) kann durch die Kombination der Beschleunigungs-Techniken **C**, **D** und **E** (13^3 Voxel) weiter auf 36% reduziert werden (vgl. Tabelle 3.4). Die Technik **C** führt alleine zu einer deutlichen Aufwandsreduktion auf 53%. Für ca. die Hälfte der Transporte brauchen keine Sichtbarkeitsberechnungen durchgeführt werden, da die korrespondierenden Patch-Paare gegenseitig vollständig sichtbar oder verdeckt sind. Beim Einsammeln von indirekter Beleuchtung führen die Beschleunigungs-Techniken **C**, **D** und **E** in diesem Beispiel zu keiner weiteren deutlichen Aufwandsreduktion. Beispielsweise werden durch die Technik **C** nur rund 17% der Transporte von den Sichtbarkeitsberechnungen ausgeschlossen. Ferner wirkt sich ein Blockierer-Cache bei im Durchschnitt nur rund 0.6 Schattenstrahlen pro Transport kaum aus. In der Detail-Szene führen die Beschleunigungs-Techniken **C** und **D** auch beim Einsammeln von indirekter Beleuchtung zu einer deutlichen Aufwandsreduktion.

C	D	E	relativer Zeitaufwand (in %)
			100
×			53
	×		68
		×	86
×	×		39
×		×	46
	×	×	62
×	×	×	36

Tabelle 3.4 Auswirkung der Beschleunigungs-Techniken **C**, **D** und **E** (direkte Beleuchtung)

Der Zeitbedarf für die Sichtbarkeitsberechnungen zur Generierung der Darstellung 3.5 mit einer Auflösung von 200^2 Pixeln konnte somit durch die Kombination der Beschleunigungs-Techniken **A** - **E** auf 365 Sekunden reduziert werden.

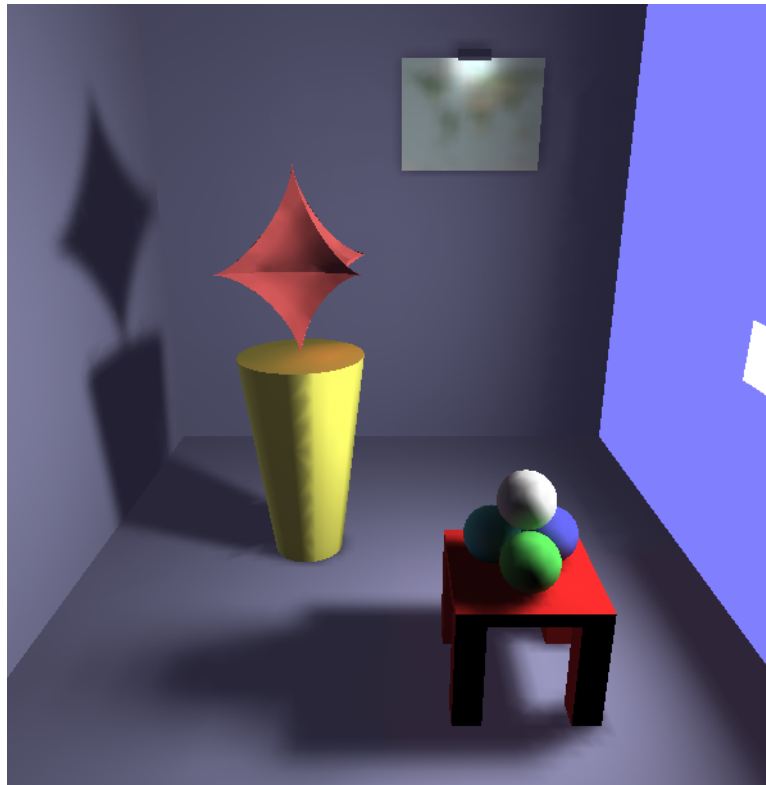


Abbildung 3.4 Gouraud-Shading

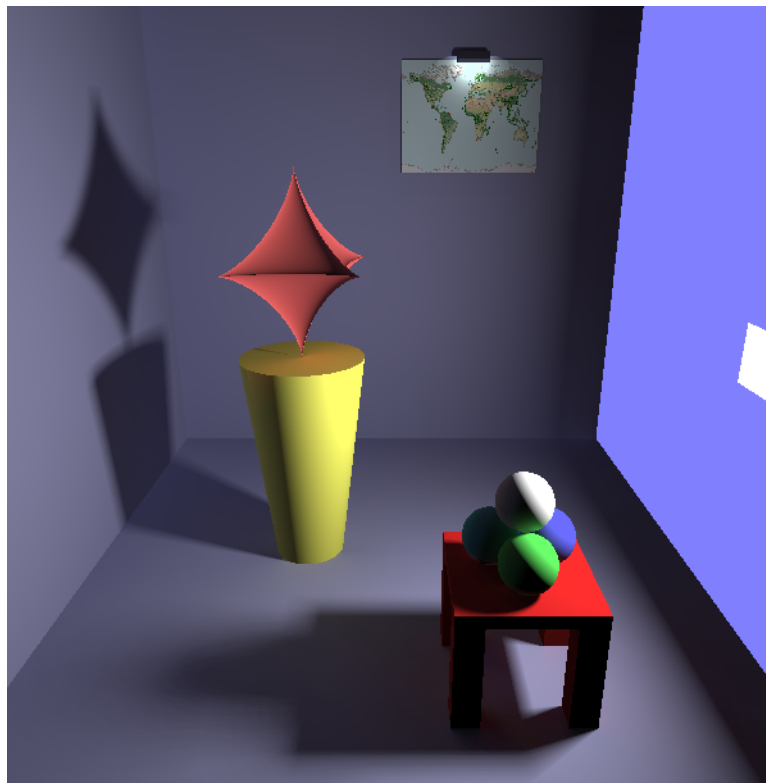


Abbildung 3.5 Final-Gathering

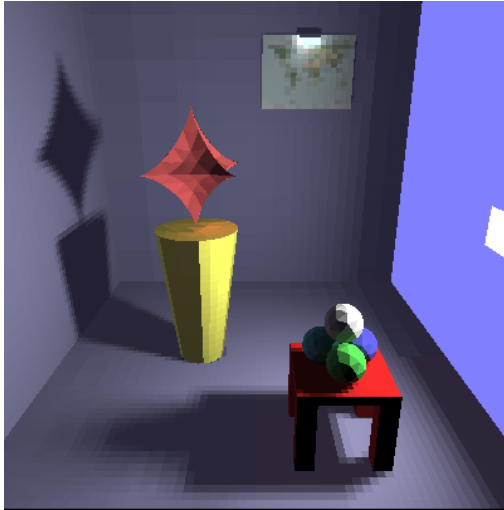


Abbildung 3.6 Keine Rekonstruktion

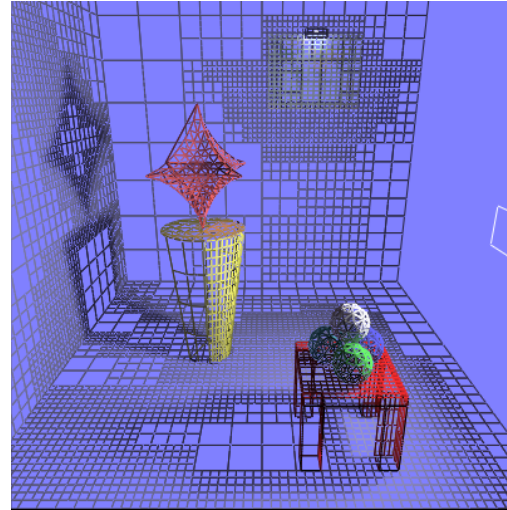


Abbildung 3.7 Meshing

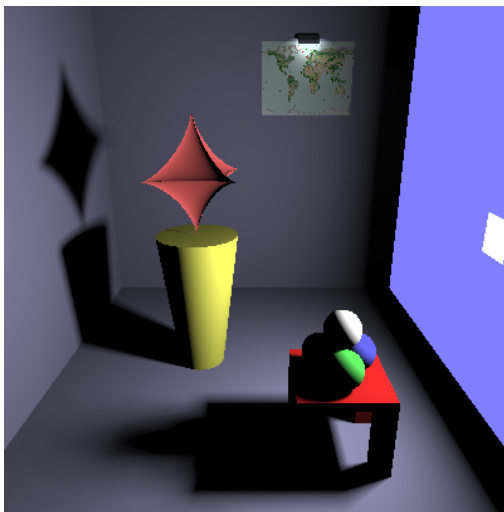


Abbildung 3.8 Direkte Beleuchtung

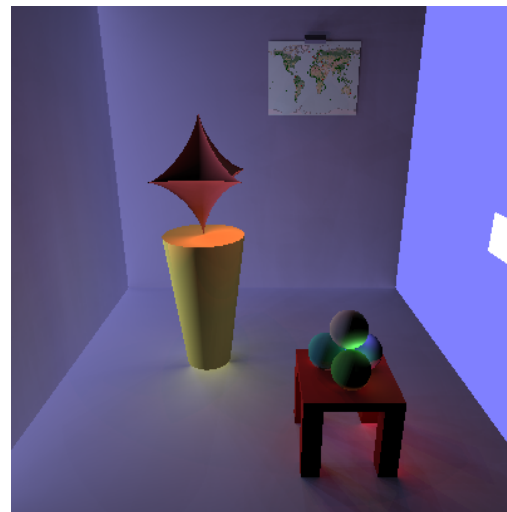


Abbildung 3.9 Indirekte Beleuchtung

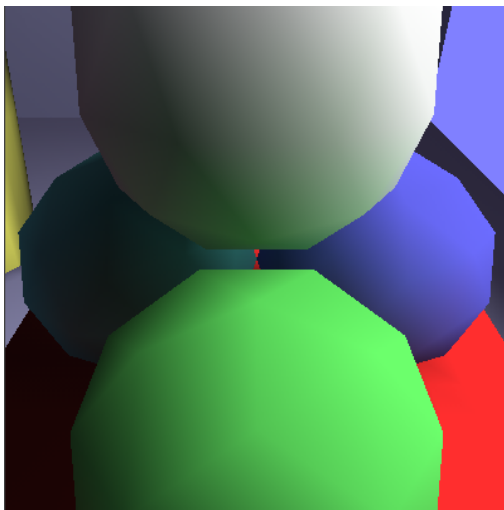


Abbildung 3.10 Gouraud-Shading

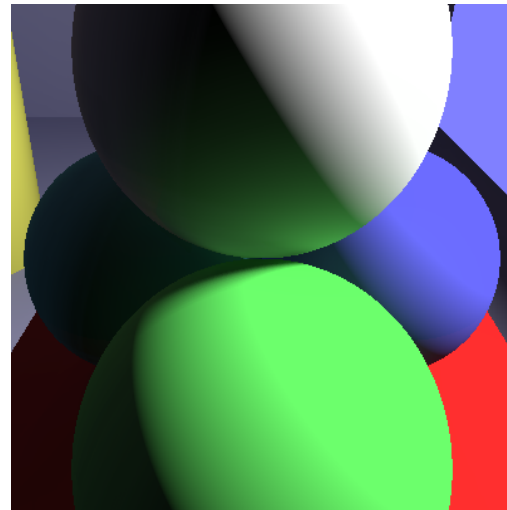


Abbildung 3.11 Final-Gathering

Kapitel 4

Wavelet-Radiance

In dem vorigen Kapitel wurden die vorwiegend praktischen Aspekte eines Wavelet-basierten globalen Beleuchtungsalgorithmus für den Radiosity-Fall beschrieben. Nun steht der allgemeine Radiance-Fall im Vordergrund. Der Aufbau dieses Kapitels ähnelt dem Aufbau des vorigen Kapitels. Die Beschreibung der Implementierung des Basis-Algorithmus und seiner Erweiterungen beschränkt sich auf die im Vergleich zum Radiosity-Fall modifizierten bzw. zusätzlichen praktischen Aspekte.

4.1 Implementierung des Basis-Algorithmus

4.1.1 Datenstrukturen

4.1.1.1 Verwaltung der Radiance-Koeffizienten

Die vierdimensionale richtungsabhängige Radiance-Verteilung wird über jedem Ausgangs-Patch bzgl. einer dynamisch wachsenden Nicht-Standard-Haar-Basis repräsentiert. Diese Darstellung läßt sich – mit Ausnahme des obersten Levels – durch eine einfache *16-tree*-Datenstruktur realisieren, die sich an der Eltern-Kind-Beziehung orientiert:

- Die Wurzel verwaltet den Koeffizienten der Mutter-Skalierungsfunktion. Die Wurzel hat entweder kein Kind oder genau ein Kind.
- Ein von der Wurzel verschiedener Knoten verwaltet die Koeffizienten von 15 Wavelets mit identischem Träger. Ein derartiger Knoten hat entweder kein Kind oder genau 16 Kinder.

Die Grundidee hinsichtlich einer einfachen Anbindung an das MRT besteht darin, zu jedem Ausgangs-Patch *simultan* zwei hierarchische Datenstrukturen zu verwalten, die eng miteinander gekoppelt sind. Abbildung 4.1 veranschaulicht die Idee im übersichtlicheren zweidimensionalen *flatland*-Fall. In *flatland* hat Radiance nur einen Positionsparameter und nur einen Richtungsparameter. Positionsträger werden durch Liniensegmente veranschaulicht und Richtungsträger durch Kreisbögen:

Radiance-Hierarchie Die Radiance-Hierarchie ist eine *16-tree*-Datenstruktur. Sie dient zur Verwaltung der Radiance-Koeffizienten gemäß dem Konzept. Analog zum Radiosity-Fall werden die beiden obersten Level der konzeptionellen Datenstruktur verschmolzen und ein zusätzlicher Level angehängt. Die Knoten fungieren als Endpunkte von Links. Innerhalb eines Levels existieren mehrere¹ Knoten mit identischem Positionsträger. Die Positionsträger werden separat in der Patch-Hierarchie verwaltet. Jeder Knoten verweist daher auf

¹ bis zu 4^{level}

Der Koeffizient der Mutter-Skalierungsfunktion entspricht der mittleren Radiance, die von dem Ausgangs-Patch in die komplette Hemisphäre ausstrahlt. Die 16 Kinder eines Knotens der Radiance-Hierarchie korrespondieren jeweils zu einem Viertel des ursprünglichen Positionsträgers und zu einem Viertel des ursprünglichen Richtungsträgers. Die Partitionierung von Positionsträgern erfolgt analog zum Radiosity-Fall. Die Partitionierung von Richtungsträgern wird im Abschnitt 4.1.5.2 beschrieben. Ein Skalierungsfunktion-Koeffizient entspricht der mittleren Radiance, die von dem Positionsträger in den Richtungsträger ausstrahlt. Es gibt genau 16 Möglichkeiten die vier Abweichungsarten² im zweidimensionalen Fall zu Abweichungsarten im vierdimensionalen Fall zu kombinieren. Ein Wavelet-Koeffizient gibt die Abweichung der mittleren Radiance über acht der 16 Teilträger von der mittleren Radiance über dem gesamten Träger gemäß einer der 15 *echten* Abweichungsarten an.

Die Patch-Hierarchie dient als Schnittstelle zum MRT. Vor der graphischen Ausgabe wird jedem Blatt der Patch-Hierarchie eine gemäß der Sichtrichtung geeignete mittlere Radiance von einem Blatt der korrespondierenden Radiance-Hierarchie zugeordnet. Bei der Berechnung der Transportkoeffizienten wird ferner auf die für Patches bzw. Patch-Paare bereits implementierten Sampling-Methoden zurückgegriffen.

4.1.1.2 Verwaltung der Transportkoeffizienten

Die Transportkoeffizienten werden analog zum Radiosity-Fall in Links verwaltet. Im Radiance-Fall verbindet ein Link jedoch nicht zwei Patches, sondern zwei Radiance-Hierarchie-Knoten. Ferner verwalten Links vom Typ LT_WS, LT_SW und LT_WW 15 Einträge der Transportmatrix anstelle von drei Einträgen. Zwischen einem geordneten Knoten-Paar können bis zu 15 Links vom Typ LT_WW existieren. In dem folgenden Pseudo-Code sind die wesentlichen von einem Link verwalteten Daten aufgeführt:

```
class t_WLLink {
    // ===== management =====
    t_WLNode* senderNode;           // pointer to sender
    t_LinkType type;                // LT_SS, LT_WS, LT_SW, OR LT_WW
    unsigned waveletIndex;          // index of sending wavelet (0,...,14)
                                    // - only for links of type LT_WW
    t_WLLink* nextLink;            // pointer to next link in the list
    // ===== transportcoefficients (TCs) =====
    t_Color cT;                    // TC between (local) scaling-functions
                                    // - part of transportmatrix only for
                                    // links of type LT_SS
    t_Color cTArray[15];           // array of TCs between basisfunctions
                                    // - only for links of type LT_WS,
                                    // LT_SW, or LT_WW
    // ===== other data =====
    t_Real kernelVariation;        // used by oracle
};
```

4.1.2 Berechnung der Transportkoeffizienten

Ein Transportkoeffizient ist als sechsdimensionales Integral über dem Produkt des Kernels mit den beiden zugehörigen Basisfunktionen definiert (vgl. Seite 18). Die Berechnung der Transportkoeffizienten im Radiance-Fall unterscheidet sich hauptsächlich hinsichtlich der folgenden Aspekte von der Berechnung der Transportkoeffizienten im Radiosity-Fall:

² „horizontal“, „vertikal“, „diagonal“ und „keine“

Berücksichtigung des Richtungsträgers des Senders Die Berechnung des Integranden unterbleibt für Samples $(x, x') \in \text{supp}_p(B_r) \times \text{supp}_p(B_s)$ mit $x - x' \notin \text{supp}_d(B_s)$. Das Vorzeichen von B_s an der Stelle $(x', x - x')$ hängt zusätzlich davon ab, in welchen Teil-Richtungsträger $x - x'$ fällt.

Berücksichtigung des Richtungsträgers des Empfängers Der Integrationsbereich wird erweitert zu $\text{supp}_p(B_r) \times \text{supp}_p(B_s) \times \text{supp}_d(B_r)$. Das Sampling von Patch-Paaren erfolgt analog zum Radiosity-Fall. Der Richtungsträger des Empfängers wird stets regulär gesampelt. Alle Samples bzgl. Patch-Paaren werden mit allen Richtungs-Samples zu Stützstellen im Radiance-Fall kombiniert. Das Vorzeichen von B_r an der Stelle (x, \vec{d}) hängt zusätzlich davon ab, in welchen Teil-Richtungsträger \vec{d} fällt.

Die Anzahl der Richtungs-Samples wird mit zunehmendem Level des Empfänger-Knotens reduziert:

$$\#RichtungsSamples = 4 \times \max(\maxPar - level, \minPar)^2$$

Die Dichte der Richtungs-Samples nimmt somit für $\maxPar \in \{3, \dots, 6\}$ und $\minPar \in \{1, 2\}$ während der ersten $\maxPar - \minPar$ Verfeinerungsschritte bzgl. des Empfängers in jedem Schritt „nur“ um einen Faktor $\in [1, 2.8]$ zu. Eine konstante Anzahl von Richtungs-Samples entspricht dem Faktor 4. Die Reduktion der Anzahl der Richtungs-Samples führt zu einer deutlichen Reduktion des Zeitaufwands ohne die Qualität der Lösung wahrnehmbar zu beeinflussen. Insbesondere wird durch diese Vorgehensweise die Verwendung von relativ vielen Richtungs-Samples auf dem obersten Level praktikabel. Ein dichtes Sampling der Hemisphäre erhöht die Wahrscheinlichkeit, alle signifikanten *glossy peaks* aufzuspüren.

Berechnung der BRDF Für die Berechnung eines Kernel-Wertes genügt im Radiosity-Fall der Zugriff auf die für jeden Oberflächenpunkt konstante BRDF. Im Radiance-Fall muß die BRDF in Abhängigkeit von der Reflexions- und von der Einstrahlrichtung gemäß dem Reflexionsmodell berechnet werden.

Insbesondere wird die Sichtbarkeitsfunktion analog zum Radiosity-Fall berücksichtigt. Der folgende Pseudo-Code beschreibt das Grundgerüst der Berechnungs-Methode:

```
void t_WLLink::computeTCs(t_WLNode* receiverNode) {
    receiverPatch = receiverNode->posSupport;
    senderPatch   = senderNode->posSupport;
    // get positional samples on receiver
    // - type==LT_W* ==> samples are ordered according to subpatches
    rPosSamples[] = receiverPatch->getSamples();
    // get positional samples on sender
    // - type==LT_W* ==> samples are ordered according to subpatches
    sPosSamples[] = senderPatch->getSamples();
    // get directional samples on receiver (ordered according to subregions)
    rDirSamples[] = receiverNode->getDirSamples();

    // integrate
    areaFraction = senderPatch->area / (nSamples * nDirSamples);
    for (i=0; i<nSamples; i++) {
        // get indices of positional samples according to sampling-strategy
        (rIndex, sIndex) = getIndices(i);
        // visibility-in-quadrature
        determine visibility between rPosSamples[rIndex] and
                                sPosSamples[sIndex]

        if (visible) {
            dirSR = rPosSamples[rIndex]-sPosSamples[sIndex];
        }
    }
}
```


Bei der Verfeinerung von Links vom Typ `LT_WS`, `LT_SW` oder `LT_WW` werden 15 bzw. 16 neue Links erzeugt (vgl. Tabelle 3.2, Seite 37). Die dabei anfallenden Sichtbarkeitsberechnungen werden durch die Einführung eines Sichtbarkeits-Caches um den Faktor 15 bzw. 4 beschleunigt³. Im Fall der Nicht-Standard-Zerlegung ist das Sichtbarkeits-Caching in [Sch96a] dokumentiert. Diese Zeit-Optimierung basiert auf der Beobachtung, daß alle 15 bzw. jeweils vier der 16 neuen Links bzgl. desselben Patch-Paares eingerichtet werden. Sie unterscheiden sich im letztgenannten Fall lediglich im Richtungsträger des Empfängers bzw. Senders. Die Richtungsträger haben jedoch keinen Einfluß auf die Sichtbarkeitsberechnung. Der Sichtbarkeits-Cache kann für jedes Sample bzgl. eines Patch-Paares den korrespondierenden bool'schen Sichtbarkeitswert aufnehmen.

4.1.5 Zusätzliche Aspekte

4.1.5.1 Reflexionsmodell

Die Berechnung eines Transportkoeffizienten basiert u. a. auf einer diskreten Menge von Werten der BRDF des Empfängers. Die BRDF wird durch eine effizient zu berechnende Funktion approximiert. Für den Einsatz in einem Radiance-Algorithmus eignet sich grundsätzlich jedes physikalisch plausible Reflexionsmodell. Praktikabel sind jedoch nur sehr einfache Modelle. Im Rahmen dieser Arbeit wird ein *isotropes* Reflexionsverhalten vorausgesetzt, d. h. die Reflexion an einem Patch sei stets invariant gegenüber einer Drehung des Patches um seine Normale. Gebürstetes Metall weist beispielsweise ein nicht-isotropes Reflexionsverhalten auf. [Sch94a] enthält einen Überblick über verschiedene Reflexionsmodelle. Christensen verwendet in seiner Implementierung das physikalisch plausible Reflexionsmodell von Ward [War92], da es einen guten Kompromiß zwischen physikalischer Genauigkeit und Effizienz darstellt. Im Rahmen dieser Arbeit wird eine Modifikation des klassischen Phong'schen Schattierungsmodells zu einem physikalisch plausiblen Reflexionsmodell verwendet [LW94]. Dieses Modell ist im Vergleich zum Ward'schen Modell einfacher und somit geringfügig effizienter. Die visuellen Ergebnisse sind vergleichbar. Die BRDF wird dabei wie folgt approximiert:

$$f_r(x, \vec{\omega}_r, \vec{\omega}_i) \approx k_d \frac{1}{\pi} + k_g \frac{n+2}{2\pi} (\cos \alpha)^n$$

Die Approximation ist als Linearkombination eines diffusen Anteils und eines glossy Anteils mit $k_d, k_g \geq 0$, $k_d + k_g \leq 1$ definiert. α bezeichnet den Winkel zwischen der Reflexionsrichtung und der gespiegelten Einstrahlrichtung (vgl. Abbildung 1.1(b), Seite 7). Das aus einer Einstrahlrichtung auf einen Oberflächenpunkt einfallende Licht wird in einen Richtungs-Kegel mit der gespiegelten Einstrahlrichtung als Achse reflektiert. Die Reflexion ist zu dieser Achse symmetrisch. Die größte Reflexion erfolgt in die gespiegelte Einstrahlrichtung und die Reflexion nimmt stetig zum Kegelrand hin ab. Die Stärke der Abnahme hängt von dem oberflächenspezifischen *Glätte-Parameter* $n \in [1, \infty[$ ab. Je glatter eine Oberfläche ist, desto schärfer fallen die glossy Reflexionen und die korrespondierenden Glanzlichter auf der Oberfläche aus.

4.1.5.2 Partitionierung von Richtungsträgern

Die Erweiterung einer Radiance-Hierarchie ist mit einer Partitionierung des Positionsträgers *und* mit einer Partitionierung des Richtungsträgers in jeweils vier Partitionen vergleichbarer Größe verbunden. Die einfache Partitionierung von Positionsträgern wurde bereits im Abschnitt 3.1.1.1 skizziert. In diesem Abschnitt wird die Partitionierung von Richtungsträgern beschrieben. Richtungen in die Hemisphäre werden in diesem Zusammenhang mit Punkten auf der korrespondieren-

³ Die Faktoren beziehen sich auf die Situation, daß Sichtbarkeit tatsächlich berechnet und nicht vererbt wird.

den Einheitshalbkugel identifiziert. Die Grundidee besteht darin, mittels der Inversen einer geeigneten Bijektion die Partitionierung eines Richtungsträgers auf die reguläre Partitionierung eines Teilquadrats des Einheitsquadrats zu *reduzieren*. Die Bijektion transformiert zusammenhängende Bereiche auf der Einheitshalbkugel in zusammenhängende Bereiche des Einheitsquadrats, wobei die Größenverhältnisse der Flächen in etwa gleich bleiben. Die Bijektion und ihre Inverse sind relativ effizient berechenbar. Die Bijektion $f : \Omega \longleftrightarrow [0, 1]^2$ ist definiert als Komposition von bijektiven Teilfunktionen (vgl. Abbildung 4.2):

$$f \equiv t \circ s \circ g$$

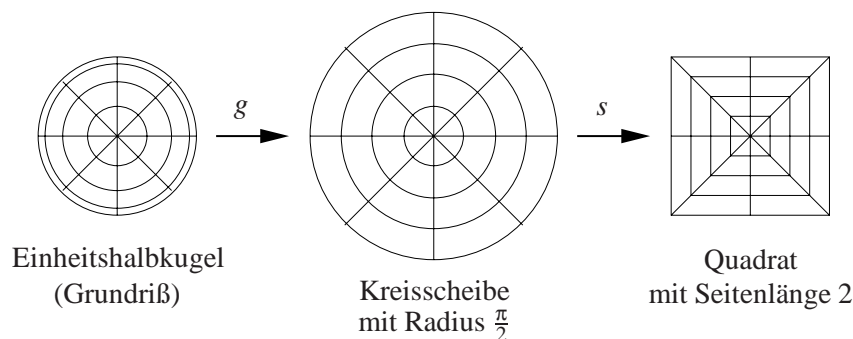


Abbildung 4.2 Bijektion $s \circ g$

Die einzelnen Teilfunktionen haben dabei die folgende Bedeutung:

- Die *Gnomonische Projektion* g bildet die Einheitshalbkugel auf die Kreisscheibe mit Radius $\frac{\pi}{2}$ ab. Geht man auf beiden Seiten von einer Polardarstellung aus, erfordert diese Abbildung *keinen* Berechnungsaufwand. Der Punkt $(1, \theta, \phi)$ wird auf den Punkt (θ, ϕ) abgebildet. Der Winkel θ bzgl. der Patch-Normalen entspricht im zweidimensionalen System dem Radius und der Winkel ϕ in der Patch-Ebene (*azimuth*) dem Polarwinkel. Folglich werden Längengrade unter Beibehaltung der Bogenlänge auf gerade Strecken durch den Ursprung abgebildet und Breitengrade unter Zunahme⁴ der Bogenlänge auf konzentrische Kreise um den Ursprung.
- Der *Radial Stretch* s bildet die Kreisscheibe mit Radius $\frac{\pi}{2}$ auf das Quadrat mit Seitenlänge 2 ab. Der Polarwinkel bleibt identisch und der Radius wird mit dem Faktor $\frac{2}{\pi \max(|\cos \phi|, |\sin \phi|)}$ skaliert. Konzentrische Kreise werden auf konzentrische Quadrate abgebildet. Die stärkste Verzerrung tritt entlang den Diagonalen auf. Sie bewegt sich jedoch noch in einem akzeptablen Rahmen.
- t ist definiert als die Komposition der uniformen Skalierung mit Faktor $\frac{1}{2}$ und der anschließenden Translation um $(\frac{1}{2}, \frac{1}{2})$. Das Quadrat mit Seitenlänge 2 um den Ursprung wird auf das Einheitsquadrat abgebildet.

Die reguläre Partitionierung des Einheitsquadrats korrespondiert zu einer Partitionierung der Einheitshalbkugel in Einheitsachtelkugeln. Abbildung 4.3 veranschaulicht die zu einer *vollständigen* regulären Partitionierung eines Viertels des Einheitsquadrats korrespondierende vollständige Partitionierung einer Einheitsachtelkugel. Entlang den Diagonalen entstehen dreieckige, ansonsten viereckige Partitionen. Die Partitionen sind in etwa gleich groß. In der Praxis erfolgt i. a. keine vollständige Partitionierung. Statt dessen werden nur die Umgebungen von gespiegelten Einstrahlrichtungen fein unterteilt, die zu signifikanten Lichttransporten korrespondieren.

⁴ Faktor $\frac{\theta}{\sin \theta} \in]1, \frac{\pi}{2}]$ für $\theta \in]0, \frac{\pi}{2}]$

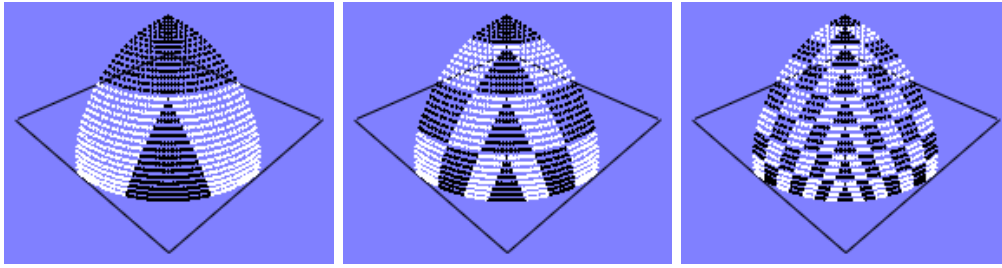


Abbildung 4.3 Vollständige Partitionierung einer Einheitsachtelkugel nach einem, zwei bzw. drei Unterteilungsschritten

Mit jedem Ausgangs-Patch wird ein lokales (u, v, w) -Koordinatensystem assoziiert. Die Normale des Patches bildet die v -Achse. Die Lage der u -Achse innerhalb der Patch-Ebene ist bei Verwendung eines isotropen Reflexionsmodells beliebig. Die u -Achse wird an der ersten Patch-Kante ausgerichtet. Die Lage der w -Achse ergibt sich aus der Forderung, daß das Patch-Koordinatensystem ein Rechtssystem sein soll. Der Ursprung ist nicht fixiert. Um effektiv globale Richtungen im Welt-Koordinatensystem in lokale Richtungen in einem Patch-Koordinatensystem konvertieren zu können und umgekehrt, werden zu jedem Ausgangs-Patch zwei zueinander inverse Transformationsmatrizen verwaltet. Abschließend soll das Zusammenspiel der Konvertierung und der Bijektion bzw. der inversen Bijektion und der inversen Konvertierung bei der Berechnung der Transportkoeffizienten beschrieben werden. Die Richtungsträger werden von den Knoten einer Radiance-Hierarchie in dem 2D-System verwaltet⁵:

Berücksichtigung des Richtungsträgers des Senders Seien x' bzw. x Samples auf dem Sender- bzw. Empfänger-Patch. Die Richtung $x - x'$ wird von dem Welt-Koordinatensystem über das Patch-Koordinatensystem des Senders in das 2D-System transformiert. Der Bereichstest läuft in dem 2D-System auf einen einfachen Punkt-in-Rechteck-Test hinaus.

Berücksichtigung des Richtungsträgers des Empfängers Das Sampling des Richtungsträgers des Empfängers wird im 2D-System durchgeführt. Ein Richtungs-Sample wird von dem 2D-System über das Patch-Koordinatensystem des Empfängers in das Welt-Koordinatensystem transformiert. Die Berechnung der BRDF basiert auf globalen Richtungen.

4.1.5.3 Extraktion einer sichtabhängigen Lösung

Der Basis-Algorithmus berechnet eine sichtunabhängige Lösung. Die Blätter einer Radiance-Hierarchie enthalten nach Ausführung des top-down Pyramidenalgorithmus die Koeffizienten dieser Lösung bzgl. der lokalen Skalierungsfunktionen. Die graphische Ausgabe erfordert die Extraktion einer sichtabhängigen Lösung zu dem vorgegebenen Augpunkt. Jedem Blatt einer Patch-Hierarchie sollte die mittlere von dem Blatt in Richtung Augpunkt ausstrahlende Radiance zugeordnet werden. Im Rahmen dieser Arbeit wurden zwei verschiedene Extraktions-Varianten implementiert.

Bei der ersten Variante wird für jedes Blatt einer Patch-Hierarchie eine eigene Sichtrichtung berücksichtigt. Eine derartige Sichtrichtung ist definiert als die Richtung vom Zentrum des Blatts zum Augpunkt. Der Radiance-Wert wird nach einer bottom-up Strategie unter Verwendung der zu jedem Patch verwalteten Knoten-Liste extrahiert. Falls die Sichtrichtung in dem Richtungsträger eines der zu dem Patch korrespondierenden Knoten der Radiance-Hierarchie liegt, entspricht dessen Skalierungsfunktion-Koeffizient dem gesuchten Radiance-Wert. Andernfalls wird die Anfrage

⁵ Ursprung und Seitenlänge des quadratischen Bereichs

rekursiv an den Vater-Patch weitergereicht⁶. Die Rekursion bricht spätestens auf Höhe der Wurzel ab, da der Richtungsträger der Wurzel einer Radiance-Hierarchie der kompletten Hemisphäre entspricht.

Bei der zweiten Variante wird für jedes Blatt einer Patch-Hierarchie dieselbe Sichtrichtung berücksichtigt. Eine derartige Sichtrichtung ist definiert als die Richtung vom Zentrum des *Ausgangs*-Patches zum Augpunkt. Der Radiance-Wert wird nach einer top-down Strategie extrahiert. Die zu der Sichtrichtung korrespondierende Teil-Hierarchie der Radiance-Hierarchie wird vollständig traversiert. Für jedes besuchte Blatt wird der Skalierungsfunktions-Koeffizient dem korrespondierenden Patch als Radiance-Wert zugeordnet. Falls dieser Patch kein Blatt ist⁶, wird der Wert an die Blätter durchgereicht.

Die erste Variante ist realitätsnäher, da die Sichtrichtung über einem Ausgangs-Patch stark variieren kann. Dies ist genau dann der Fall, wenn sich der Augpunkt relativ zur Größe des Ausgangs-Patches dicht über dem Ausgangs-Patch befindet. Außerdem erfolgt ein kontinuierlicher Übergang zwischen zwei benachbarten Radiance-Hierarchien. Schließlich reagieren die Glanzlichter „sensibler“ auf eine Änderung des Augpunktes. Die zweite Variante führt hingegen zu „symmetrischeren“ Glanzlichtern auf einem Ausgangs-Patch. Ferner ist diese Variante aufgrund der top-down Strategie effizienter. Mittels einer Programm-Option kann zwischen beiden Varianten gewählt werden.

4.1.6 Anbindung an das MRT

Die Anbindung des Basis-Algorithmus an das MRT erfolgt analog zum Radiosity-Fall. Eine zentrale Voraussetzung für diese einfache Anbindung ist die simultane Verwaltung von Patch-Hierarchien zu den Radiance-Hierarchien. Die Patch-Hierarchien dienen als Schnittstelle zum MRT.

4.2 Erweiterungen

4.2.1 Sonderbehandlung von diffusen Reflektoren

In vielen Szenen übersteigt der Anteil der geometrischen Objekte mit eher diffusem Reflexionsverhalten deutlich den Anteil der Objekte mit stark ausgeprägtem glossy Reflexionsverhalten. In diesen Fällen lassen sich Zeit- und Speicheraufwand durch die Sonderbehandlung von diffusen Reflektoren deutlich reduzieren [Sch94b].

Bei der Implementierung werden zwei Typen von Radiance-Hierarchien unterschieden: diffuse Radiance-Hierarchien und glossy Radiance-Hierarchien. Über einem diffus reflektierenden Ausgangs-Patch wird anstelle einer vierdimensionalen Radiance-Verteilung „nur“ eine zweidimensionale diffuse Radiance-Verteilung analog zum Radiosity-Fall verwaltet. Die Klasse zur Verwaltung eines Knotens der Radiance-Hierarchie `t_WLNode` wurde in eine abstrakte Basis-Klasse umgewandelt, von der zwei neue Klassen `t_WLNodeD` und `t_WLNodeG` abgeleitet wurden. Bei den Transporten zwischen Basisfunktionen können die folgenden vier Fälle auftreten: diffus \leftarrow diffus, diffus \leftarrow glossy, glossy \leftarrow diffus und glossy \leftarrow glossy. Transporte zwischen Basisfunktionen „vom gleichen Typ“ wurden bereits diskutiert. Transporte zwischen Basisfunktionen „verschiedenen Typs“ lassen sich leicht ableiten. Beim Transport zu einem diffusen Empfänger entfallen die Integration über den Richtungsträger des Empfängers und die Berechnung der BRDF. Beim Transport von einem diffusen Sender entfällt die Überprüfung, ob die Einstrahlrichtungen im Richtungsträger des Senders liegen. Von den Klassen `t_WLLinkSW`, `t_WLLinkWS`

⁶ Abbildung 4.1 auf Seite 53 zeigt eine derartige Situation. Der Augpunkt befindet sich rechts von dem Ausgangs-Patch.

und t_{WLinkWW} wurden jeweils vier neue Klassen zur Verwaltung entsprechender Transportkoeffizienten abgeleitet. Die Extraktion einer sichtabhängigen Lösung erfolgt nur für glossy reflektierende Objekte. In der Initialisierungs-Methode wird dazu eine Liste von Zeigern auf alle glossy reflektierenden Objekte angelegt. Der derart erweiterte Wavelet-Radiance-Algorithmus beleuchtet insbesondere diffuse Szenen ähnlich⁷ effizient wie der Wavelet-Radiosity-Algorithmus.

4.2.2 Final-Gathering

Das Final-Gathering wurde mit wenigen Ausnahmen analog zum Radiosity-Fall implementiert. Insbesondere orientiert sich das Einsammeln von Radiance ebenfalls *indirekt* an der Link-Datenstruktur. Für das Markieren der relevanten Sender-Teil-Hierarchien werden alle Knoten der Empfänger-Radiance-Hierarchie entlang des Pfades von dem mit $(x, \vec{\omega})$ assoziierten Blatt-Knoten zu der Wurzel besucht und die korrespondierenden Listen von Links traversiert.

Das Einsammeln von Radiance erfolgt im Radiance-Fall mittels der wie folgt definierten Punkt-Basisfunktion-Transportkoeffizienten:

$$T_{(x, \vec{\omega}) \leftarrow s} \equiv \int_{x' \in \text{supp}_p(B_s)} f_r(x, \vec{\omega}, x' - x) g(x, x') B_s(x', x - x') dx'$$

Im Vergleich zu den gewöhnlichen Transportkoeffizienten zwischen zwei Basisfunktionen entfällt die Integration über den Träger einer Empfänger-Basisfunktion. Statt dessen bezieht sich der Transportkoeffizient ausschließlich auf die Differential-Fläche dx und die Sichtrichtung $\vec{\omega}$. Die unverdeckten Punkt-Basisfunktion-Transportkoeffizienten werden durch eine entsprechend vereinfachte Methode zur Berechnung von gewöhnlichen Transportkoeffizienten berechnet.

4.3 Ergebnisse

In diesem Abschnitt werden die Ergebnisse von einigen Wavelet-Radiance-Berechnungen für zwei sehr einfache Szenen dokumentiert. Zunächst liegt das Hauptaugenmerk auf der direkten Beleuchtung. Anschließend wird kurz das Einsammeln von indirekter Beleuchtung über ein glossy reflektierendes Objekt durch Final-Gathering untersucht. Die Berechnungen wurden auf einer SGI Indigo 2 (R4400/200 MHz) durchgeführt.

Die erste Szene besteht aus zwei nebeneinander angeordneten Quadraten mit unterschiedlichem Reflexionsverhalten, die direkt von drei verschiedenfarbigen Lichtquellen beleuchtet werden (vgl. Abbildung 4.7). Das linke Quadrat ist glossy reflektierend und das rechte Quadrat ist diffus reflektierend. Die Lichteinstrahlung auf das glossy reflektierende Quadrat wird durch ein (diffus reflektierendes) Dreieck blockiert. Es werden vier Verfeinerungsschritte durchgeführt.

Die Abbildung 4.4 zeigt die berechnete approximative Lösung *ohne* den Einsatz einer Rekonstruktionsmethode *für einen bestimmten Augpunkt*. Der interessante Aspekt besteht darin, daß die berechnete Radiance-Verteilung *sichtunabhängig* ist. Bei einer Änderung des Augpunktes braucht keine neue Radiance-Verteilung berechnet zu werden. Statt dessen genügt die Extraktion einer sichtabhängigen Radiance-Verteilung für den neuen Augpunkt (vgl. Abbildung 4.5). Das diffus reflektierende Quadrat sieht aus allen Blickrichtungen gleich aus. Demgegenüber ändert sich die Lage der Glanzlichter auf dem glossy reflektierenden Quadrat.

Eine extrahierte Radiance-Verteilung läßt sich – analog zum Radiosity-Fall – entweder effizient durch Gouraud-Shading (vgl. Abbildung 4.6) oder durch Final-Gathering (vgl. Abbildung 4.7) glätten. Die relativ kostengünstige Extraktion ermöglicht in Kombination mit dem hardwareunterstützten Gouraud-Shading ein interaktives Erkunden des Reflexionsverhaltens des glossy reflektierenden Quadrats. Die Radiance-Verteilung ändert sich dabei aufgrund der stückweise konstanten Darstellung beim Übergang zwischen zwei Teil-Richtungsträgern i. a. „sprunghaft“. Mit

⁷ Die simultane Verwaltung von Patch- und Radiance-Hierarchien führt zu einem *geringfügig* höheren Speicheraufwand.

zunehmender Verfeinerungstiefe nimmt jedoch die gerichtete Auflösung zu (vgl. Abbildung 4.3, Seite 59) und somit die Stärke der Sprünge ab. Das Final-Gathering erzeugt ggf. das folgende Artefakt: Bei einer geringen Sample-Dichte auf dem Sender und einem hohen Glätte-Parameter des glossy Reflektors sind die Spiegelbilder der einzelnen Samples deutlich zu erkennen. Ein möglicher Lösungsansatz besteht darin, die Anzahl der Samples nicht konstant zu wählen, sondern proportional zu dem vom Sender aufgespannten Raumwinkel. Der Proportionalitätsfaktor könnte ferner von dem Glätte-Parameter abhängen.

Die Reduktion der Anzahl der Richtungs-Samples führt in diesem Beispiel zu einer Reduktion des Zeitaufwands für die Berechnung der glossy Transportkoeffizienten auf 9%. Tabelle 4.2 dokumentiert die Aufwandsreduktion durch die Sonderbehandlung von diffusen Reflektoren. Der Zeitaufwand bezieht sich auf die Berechnung der Transportkoeffizienten und der Speicheraufwand auf die Verwaltung der Radiance- und Transportkoeffizienten⁸. Die deutliche Aufwandsreduktion hinsichtlich der Berechnung und Verwaltung der diffusen Radiance-Verteilung ist dadurch zu erklären, daß ein diffuses Reflexionsverhalten den *worst case* für den Radiance-Algorithmus darstellt. Für alle Positionen sind alle Reflexionsrichtungen gleich signifikant, so daß eine *vollständige* glossy Radiance-Hierarchie mit einer entsprechend großen Anzahl von Links aufgebaut wird. Der verbleibende große Speicherbedarf für die Verwaltung der glossy Radiance- und Transportkoeffizienten ist der Preis für die Berechnung einer *sichtunabhängigen* Lösung.

	ohne Sonderbehandlung von diffusen Reflektoren	mit Sonderbehandlung von diffusen Reflektoren
Zeitaufwand (in Sekunden)	42	11
Speicheraufwand (in kBytes)		
Patch-Hierarchien	204	204
diffuse Radiance-Hierarchien	–	29
glossy Radiance-Hierarchie	9462	2436
Links diffus ← diffus	–	37
Links glossy ← diffus	6842	1393
	16508	4099

Tabelle 4.2 Zeit- und Speicheraufwand (Glätte-Parameter 100)

Den bisher dokumentierten Ergebnissen liegt der Glätte-Parameter 100 zugrunde. Eine Änderung dieses Parameters wirkt sich wie folgt auf den Aufwand aus: Für die Parameter 500 (vgl. Abbildung 4.8) bzw. 20 (vgl. Abbildung 4.9) wurde ein Zeitaufwand von 6 bzw. 17 Sekunden ermittelt und ein Speicheraufwand von 2391 bzw. 6353 kBytes. Je diffuser das Reflexionsverhalten, desto größer der Aufwand. Dieses Ergebnis stützt die Aussage, daß ein diffuses Reflexionsverhalten den *worst case* darstellt. Ferner unterstreicht es die Wichtigkeit der Sonderbehandlung von diffusen Reflektoren.

Die zweite Szene ist in der Abbildung 4.10 skizziert. Eine diffus reflektierende Kugel D wird über ein kleines glossy reflektierendes Quadrat G indirekt beleuchtet. Die Abbildung 4.11 zeigt die mittels Final-Gathering von dem Mittelpunkt des glossy Reflektors rekonstruierte Radiance-Verteilung auf dem diffusen Reflektor nach drei, vier bzw. fünf Verfeinerungsschritten. Im Vergleich zum Radiosity-Fall führt das Final-Gathering nicht notwendigerweise zu einer geglätteten Lösung. Einerseits werden zwar die Unstetigkeiten an den Grenzen der Teil-Richtungsträger des Empfängers beseitigt. Andererseits treten jedoch die Unstetigkeiten an den Teil-Richtungsträgern des glossy Reflektors deutlich hervor.

⁸ Der Speicheraufwand für die Verwaltung der topologischen Datenstruktur wird dabei nicht berücksichtigt.

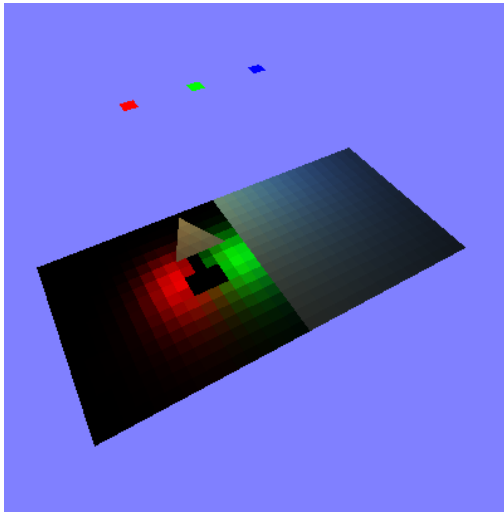


Abbildung 4.4 Erste Blickrichtung

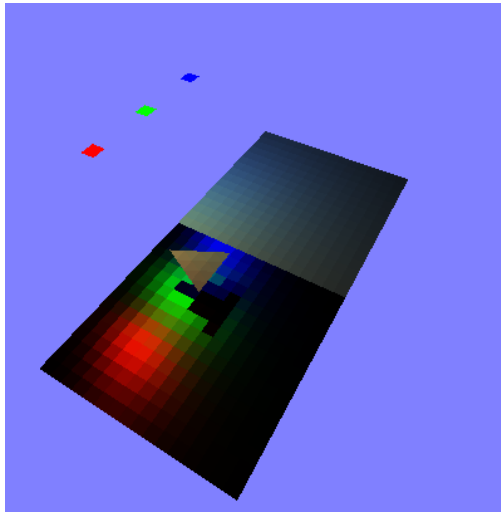


Abbildung 4.5 Zweite Blickrichtung

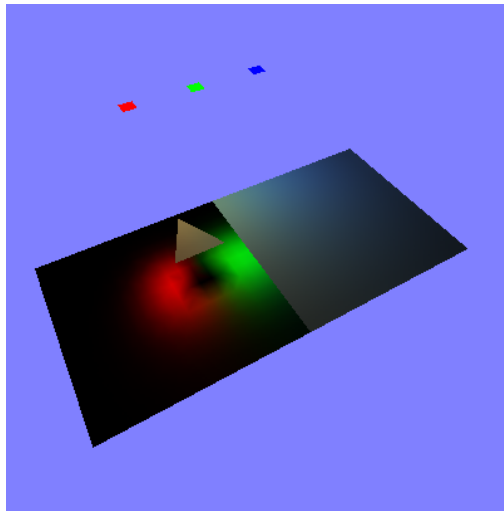


Abbildung 4.6 Gouraud-Shading

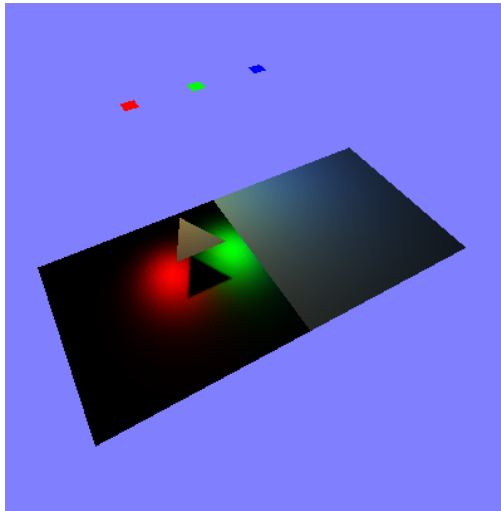


Abbildung 4.7 Final-Gathering

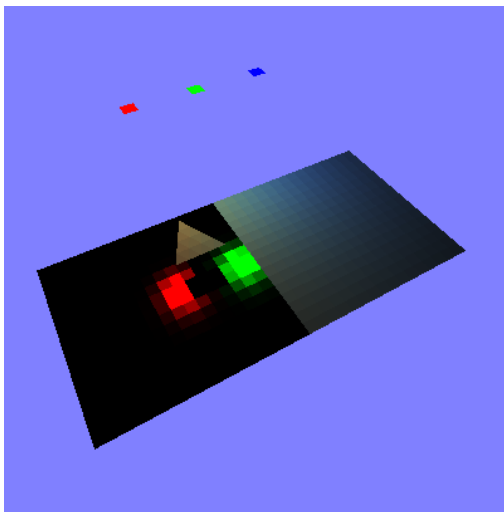


Abbildung 4.8 Glätte-Parameter 500

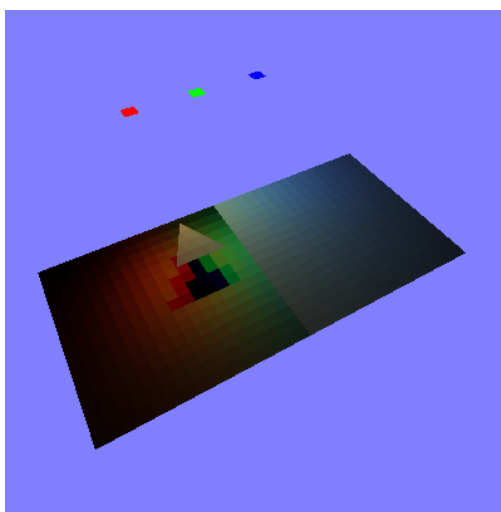


Abbildung 4.9 Glätte-Parameter 20



Abbildung 4.10 Szenengeometrie

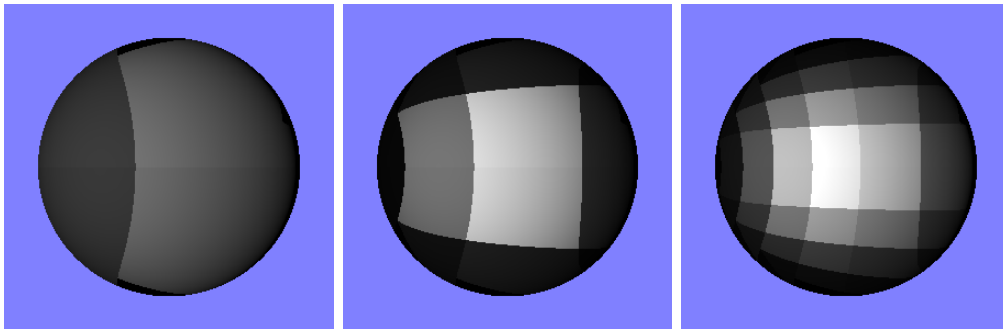


Abbildung 4.11 Final-Gathering nach drei, vier bzw. fünf Verfeinerungsschritten

Kapitel 5

Zusammenfassung und Ausblick

Finite-Elemente-Methoden zum approximativen Lösen des globalen Beleuchtungsproblems berechnen grundsätzlich eine *sichtunabhängige* Lösung, die sich für eine interaktive Ausgabe eignet. Sie profitieren im Radiosity-Fall von der im Vergleich zum allgemeinen Radiance-Fall geringeren Dimensionalität der gesuchten Radiance-Verteilung. Die Entwicklung bzw. die Anwendung von ausgefeilten Techniken hat in den vergangenen Jahren im Radiosity-Fall zu signifikanten Effizienz- bzw. Qualitätsverbesserungen geführt: u. a. Einsatz von Wavelet-Basen, clustering, importance-basierte Verfeinerung, Integration von gekrümmten Objekten, Final-Gathering. Es stellt sich die Frage, inwieweit sich diese Techniken hinsichtlich FE-Methoden für den Radiance-Fall adaptieren lassen.

Grundlage dieser Arbeit ist eine von Christensen vorgeschlagene Finite-Elemente-Methode für den Radiance-Fall. Christensen adaptiert die genannten Techniken. Der Schwerpunkt dieser Arbeit liegt – neben der Implementierung des Algorithmus auf Basis des 3D-Graphiksystems MRT – auf der Identifizierung der beim Übergang vom klassischen Radiosity-Fall zum Radiance-Fall modifizierten bzw. zusätzlichen theoretischen und praktischen Aspekte.

Bei der Aufarbeitung des gemeinsamen theoretischen Hintergrunds von Wavelet-basierten globalen Beleuchtungsalgorithmen wird an den entsprechenden Stellen auf die Unterschiede verwiesen: u. a. Dimensionalität von Radiance-Verteilung, BRDF und Kernel, Definition von Transportoperator und -koeffizienten. Von den genannten Techniken wurde der Einsatz von Wavelet-Basen und das Final-Gathering realisiert. Die Implementierung erfolgte in zwei Phasen. Zunächst wurde ein Wavelet-Radiosity-Algorithmus implementiert. Danach wurde darauf aufbauend der Wavelet-Radiance-Algorithmus realisiert. Auf diese Weise wurde das „Implementierungs-Problem“ in zwei einfachere Teilprobleme zerlegt. Ferner erleichterte diese Vorgehensweise die genaue Identifizierung der modifizierten bzw. zusätzlichen praktischen Aspekte: u. a. Grad der Hierarchien zur Verwaltung der Radiance-Koeffizienten, Berechnung der Transportkoeffizienten, Berücksichtigung eines Reflexionsmodells, Partitionierung von Richtungsträgern, Extraktion einer *sichtabhängigen* Lösung. Die im Rahmen dieser Arbeit erstellte Implementierung weist im Vergleich zu der von Christensen vorgestellten Implementierung die folgenden besonderen Merkmale auf:

- einfache Anbindung an ein existierendes Wavelet-Radiosity-System
- interaktive Ausgabe
- Sonderbehandlung von diffusen Reflektoren

Ferner konnte die visuelle Qualität der Darstellung von gekrümmten Objekten beim Final-Gathering unter Ausnutzung von Objekt-Informationen verbessert werden.

Vor dem Ausblick folgt ein kritischer Rückblick auf zwei zentrale Design-Entscheidungen:

Gerichteter Transport Der gerichtete Transport zeichnet sich gegenüber dem 3-Punkt-Transport (vgl. Seite 14) durch ein effizienteres initial linking¹ und die Berechnung einer sichtunabhängigen Lösung aus. Bei dem 3-Punkt-Transport kann die Radiance über einem Ausgangs-Patch nur in Richtung eines anderen Patches abgegriffen werden. Der Augpunkt wird üblicherweise durch einen sehr kleinen Aug-Patch modelliert. Allerdings erschwert das *Trägermismatch*² bei dem gerichteten Transport die Berechnung von sekundären Glanzlichtern, da das Aufspüren von glossy peaks der Radiance-Verteilung über sekundären Lichtquellen problematischer ist. Ein weiterer Nachteil ist der zeitliche Mehraufwand, der von den notwendigen Richtungs-Umrechnungen verursacht wird.

Die Design-Entscheidung, auf den gerichteten Transport zu setzen, hat sich jedoch insgesamt bewährt.

Standard-Zerlegung des Transportoperators Die Standard-Zerlegung des Transportoperators wird in dieser Arbeit als „ungewöhnlich“ bezeichnet, da die Gruppe um Christensen offensichtlich bisher als einzige einen Wavelet-basierten globalen Beleuchtungsalgorithmus vorgestellt hat, der auf dieser Art der Zerlegung des Transportoperators basiert. In dem Abschnitt 2.2.4.4 wird gezeigt, daß die Standard-Zerlegung des Transportoperators theoretisch einfach zu erklären ist: Die explizite Verwendung einer Wavelet-Basis bei der Diskretisierung der Radiance-Gleichung durch die Galerkin-Methode korrespondiert unmittelbar zu der Standard-Zerlegung des Transportoperators. Christensen verfolgt diese Zerlegung, da sie nach den praktischen Erfahrungen verschiedener Autoren häufig zu einer dünner besetzten Transportmatrix führt. Demgegenüber steht jedoch – neben der etwas besseren asymptotischen Komplexität – der wesentlich geringere Implementierungsaufwand eines auf der Nicht-Standard-Zerlegung des Transportoperators basierenden Algorithmus (vgl. Abschnitt 3.2). In [Sch96a] wird ein entsprechender Wavelet-Radiance-Algorithmus vorgestellt, der als das Pendant des klassischen Hierarchical-Radiosity-Algorithmus angesehen werden kann.

Die Design-Entscheidung, auf die Standard-Zerlegung des Transportoperators zu setzen, hat sich daher nicht bewährt.

Die enorme Komplexität des globalen Beleuchtungsproblems im allgemeinen Radiance-Fall hat zur Folge, daß sich trotz des Einsatzes von Wavelet-Basen mit dem implementierten Algorithmus in der Praxis nur sehr einfache Szenen beleuchten lassen. Im folgenden werden einige Erweiterungen hinsichtlich der Integration von gekrümmten Objekten sowie zur Reduktion des Zeit- und Speicheraufwands skizziert.

Integration von gekrümmten Objekten Man könnte der Frage nachgehen, ob sich die im Radiosity-Fall bewährte Integration von gekrümmten Objekten unter Ausnutzung von Objekt-Informationen auch für den Radiance-Fall adaptieren läßt. Erste Versuche ohne spezielle algorithmische Anpassungen haben ergeben, daß sich bei direkter Beleuchtung zwar Glanzlichter auf gekrümmten Objekten herausbilden. Allerdings treten deutlich wahrnehmbare Artefakte auf: Glanzlichter werden zum Teil „unsauber“ herausgebildet bzw. „verschwinden“ sogar bei bestimmten Blickrichtungen. Ein Problem liegt darin, daß mit jeder Radiance-Hierarchie „nur“ eine Hemisphäre assoziiert ist, in die Radiance ausstrahlt. Im Fall eines konvex gekrümmten Objekts vergrößert sich jedoch der Ausstrahlungsbereich mit fortschreitender Unterteilung aufgrund der Verschiebung der neu erzeugten Knoten auf die Oberfläche.

Entkoppelung von Positions- und Richtungsverfeinerung Das wohl größte Manko des vorgestellten Algorithmus ist die Koppelung von Positions- und Richtungsverfeinerung (vgl. Ab-

¹ $O(k^2)$ gegenüber $O(k^3)$ mit $k \equiv \#$ Ausgangs-Patches

² i. a. $supp_d(B_s) \not\subseteq supp_p(B_r)$, vgl. Abbildung 2.7 auf Seite 19

schnitt 4.1.4). Die Verfeinerung eines Positionsträgers geht stets mit einer Verfeinerung des Richtungsträgers einher und umgekehrt. Diese unflexible Koppelung ist eine Folge der expliziten (Standard-Zerlegung des Transportoperators) bzw. impliziten (Nicht-Standard-Zerlegung des Transportoperators) Verwaltung der Radiance-Verteilung bzgl. vierdimensionaler Nicht-Standard-Wavelet-Basen. Die Sonderbehandlung von diffusen Reflektoren ist ein erster Schritt in Richtung Entkoppelung von Positions- und Richtungsverfeinerung. Für diffuse Reflektoren werden nur Positionsverfeinerungen durchgeführt, d. h. bei einer Unterteilung entstehen anstelle von 16 nur vier neue Knoten. Analog sollten im dualen Extremfall eines sehr kleinen glossy Reflektors nur Richtungsverfeinerungen durchgeführt werden. Der allgemeine Fall wird ausführlich in [Sch96a] diskutiert. Die dort vorgestellte Idee für einen allgemeinen Lösungsansatz besteht darin, bei einer Verfeinerungsentscheidung stets alle drei Möglichkeiten – nur Positionsverfeinerung, nur Richtungsverfeinerung bzw. Positions- und Richtungsverfeinerung – in Betracht zu ziehen. Die Entscheidung könnte sich an der Variation des geometrischen Terms und an der Variation der BRDF orientieren. Das Hauptproblem dieses Ansatzes liegt darin, daß die Verfeinerungsentscheidungen hinsichtlich eines Empfänger-Knotens und verschiedener Sender-Knoten zu verschiedenen Ergebnissen kommen könnten. Man müßte Mechanismen zum Umgang mit bzgl. verschiedener Basen repräsentierten Teilen der Radiance-Verteilung entwickeln.

Sonderbehandlung des diffusen Teils einer Radiance-Verteilung Für Objekte, die eine Kombination eines signifikanten diffusen und eines signifikanten glossy Reflexionsverhaltens aufweisen, könnte man den diffusen Teil der Radiance-Verteilung gesondert behandeln. Die Implementierung müßte dahingehend abgeändert werden, daß mit einem Ausgangs-Patch ggf. *zwei* Radiance-Hierarchien assoziiert werden können: eine diffuse Radiance-Hierarchie für die Verwaltung des diffusen Teils und eine glossy Radiance-Hierarchie für die ausschließliche Verwaltung des glossy Teils der Radiance-Verteilung. Auf diese Weise ließe sich die sehr sinnvolle Sonderbehandlung von diffusen Reflektoren verallgemeinern.

Importance-basierte Verfeinerung Der Importance-Ansatz [SAS92] erweitert den beschriebenen hierarchischen Ansatz dahingehend, daß sich die Beleuchtungsrechnung auf vorgegebene *wichtige* Teilszenen fokussieren läßt. Der Algorithmus erwartet als Eingabe neben einer emittierenden Radiance-Verteilung auch eine emittierende Importance-Verteilung. Simultan zu der Radiance-Transportgleichung wird eine adjungierte Importance-Transportgleichung gelöst. Importance wird dabei in solche Bereiche transportiert, die signifikant zur Beleuchtung der vorgegebenen wichtigen Teilszenen beitragen. Das Orakel zur simultanen Verfeinerung der beiden Linearen Gleichungssysteme hängt zusätzlich von dem Betrag des Importance-Koeffizienten der Empfänger-Basisfunktion ab. Auf diese Weise wird die Radiance-Verteilung nur für wichtige Teilszenen genau berechnet und die Importance-Verteilung nur für helle Teilszenen. Dieser Ansatz eignet sich z. B. zur effizienten Berechnung einer sichtabhängigen Lösung, indem die Beleuchtungsrechnung auf die sichtbare Teilszene fokussiert wird. Der Effizienz-Gewinn fällt im Radiance-Fall im Vergleich zum Radiosity-Fall noch deutlicher aus [Chr95].

Clustering Der Clustering-Ansatz [SAG94] erweitert den hierarchischen Ansatz in bottom-up Richtung, indem lokale Ansammlungen von Objekten beim Lichttransport ggf. als Einheit betrachtet werden. Formal läuft Clustering darauf hinaus, daß die Hierarchie-Bildung auf einer höheren Stufe ansetzt. Die Hierarchie-Bildung beginnt nicht erst auf der Stufe der Ausgangs-Patches, sondern bereits auf der Stufe von groben Ansammlungen von Objekten. Mit fortschreitender Verfeinerung werden diese in immer feinere Ansammlungen aufgebrochen bis hin zur Stufe der Ausgangs-Patches. Ab dann erfolgt die Verfeinerung wie beschrieben. Der Clustering-Ansatz ist ein unentbehrliches Mittel zur Beleuchtung von komplexen Szenen, da der Aufwand nicht weiter von der Anzahl der Ausgangs-Patches abhängt.

Anhang A

Programm-Optionen

Die Buchstaben n bzw. r sind Platzhalter für numerische Parameter (natürliche bzw. reelle Zahlen). In eckigen Klammern ist jeweils die Vorbelegung angegeben.

<i>Option</i>	<i>WB/WL</i>	<i>Beschreibung</i>
Berechnung der Transportkoeffizienten		
Vgl. Abschnitte 3.1.2 bzw. 4.1.2		
nSamples n [256/16] nDirSamplesParMax n [4]	×/× -/×	Gewünschte Anzahl der Samples bzgl. eines Patch-Paares Parameter zur Steuerung der Anzahl der Richtungs-Samples bzgl. eines Empfängers. Vgl. Seite 55
nDirSamplesParMin n [1] randomSampling	-/× ×/×	dito Samples werden auf einem Patch stets gleichverteilt zufällig plaziert.
oneToOneSampling	×/×	Samples auf dem Empfänger-Patch werden mit genau einem Sample auf dem Sender-Patch zu Samples bzgl. des Patch-Paares kombiniert.
noInheritTotalVis noVisibilityCaching noSampleReductionLTSS	×/× -/× ×/×	Die optimierte Sichtbarkeitsberechnung wird deaktiviert. Das Sichtbarkeits-Caching wird deaktiviert. Vgl. Seite 56 Bei der Berechnung von Transportkoeffizienten zwischen Skalierungsfunktionen werden standardmäßig nur 25% der spezifizierten Samples bzgl. eines Patches verwendet. Die Reduktion wird nur dann durchgeführt, wenn wenigstens zwei Samples auf einem Patch verbleiben. Andernfalls wäre die Kernel-Variation bzgl. eines Patch-Paares undefiniert.
maxGeometricTerm r [50.0]	×/×	Obere Schranke für den geometrischen Term. Diese Option steuert den einfachen Mechanismus zur Vermeidung von Singularitäts-Problemen. Ein negativer Wert steht für „keine obere Schranke“.
checkSampling	×/×	Die Samples bzgl. eines Patch-Paares werden graphisch dargestellt. Im Radiance-Fall werden auch die Samples bzgl. eines Richtungsträgers graphisch dargestellt (relativ zum Patch-Zentrum).
Lösen der diskreten Radiance-Gleichungen		
Vgl. Abschnitte 3.1.3 bzw. 4.1.3		
iterationThreshold r [0.0001]	×/×	Die Iteration bricht ab, wenn sich die Farbkomponenten aller Radiance-Koeffizienten innerhalb eines Iterationsschrittes nur um einen Betrag unterhalb dieses Schwellwertes ändern.
<i>Fortsetzung auf nächster Seite</i>		

Fortsetzung von voriger Seite		
Option	WB/WL	Beschreibung
maxIterationSteps n [4] initializeSolution	×/× ×/×	Obere Schranke für die Anzahl der Iterationsschritte Standardmäßig wird der aktuelle Lösungsvektor als Startvektor für den nächsten Lösungsschritt verwendet. Der Startvektor kann alternativ gemäß der emittierenden Radiance-Verteilung initialisiert werden.
jacobiIteration	×/–	Anstelle des standardmäßig verwendeten Gauss-Seidel-Verfahrens wird das Jacobi-Verfahren eingesetzt. Diese Option wird nur dann berücksichtigt, wenn das Programm mit einer entsprechenden Compiler-Option übersetzt wurde. Das Jacobi-Verfahren erfordert zusätzlichen Speicherplatz. Vgl. Seite 19
checkSolving	×/×	Die aktuelle Lösung wird nach jedem Iterationsschritt graphisch dargestellt.
Verfeinerung der diskreten Radiance-Gleichungen		
Vgl. Abschnitte 3.1.4 bzw. 4.1.4		
significanceThreshold r [0.00001]	×/×	Absoluter Signifikanzschwellwert. Ein negativer Wert aktiviert die relative Definition des Signifikanzschwellwerts.
significanceFraction r [0.1/0.03]	×/×	Bei der relativen Definition des Signifikanzschwellwerts dient dieser Bruchteil der maximalen Signifikanz aller potentiell neuen Transportkoeffizienten als Signifikanzschwellwert.
noAdaptiveIntegration	×/×	Die adaptive numerische Integration zur genaueren Berechnung von „alten“ Transportkoeffizienten wird deaktiviert.
areaThreshold r [0.0]	×/×	Ein Patch wird nicht weiter unterteilt, falls sein Flächeninhalt diesen Schwellwert unterschreitet. Der Schwellwert kann auch szenenspezifisch in der Beschreibungs-Datei mittels dem Token AEPS definiert werden.
areaOracle	×/×	Das Signifikanzmaß umfaßt als zusätzlichen Faktor den Flächeninhalt des Empfänger-Patches.
loops n [4]	×/×	Obere Schranke für eine Folge von Verfeinerungsschritten (Vorbelegung). Nach einer Folge von Verfeinerungsschritten besteht die Möglichkeit, eine weitere Folge von Verfeinerungsschritten anzustoßen. Zu Beginn jeder Folge kann die obere Schranke interaktiv gemäß der Güte der aktuellen Lösung angepaßt werden.
Final-Gathering		
Vgl. Abschnitte 3.4.2 bzw. 4.2.2		
fgNSamplesDirIllum n [9]	–/×	#Samples pro Sender-Patch zur Ermittlung des unverdeckten Punkt-Skalierungsfunktion-Transportkoeffizienten bei direkter Beleuchtung
fgNSamplesIndIllum n [1]	–/×	dito für indirekte Beleuchtung
fgRandomSampling	–/×	Samples werden auf einem Sender-Patch stets gleichverteilt zufällig plaziert.
fgNShadowRaysDirIllum n [64]	×/×	#Schattenstrahlen zur Ermittlung der Punkt-Patch-Sichtbarkeit bei direkter Beleuchtung
fgNShadowRaysIndIllum n [64]	×/×	dito für indirekte Beleuchtung
fgNoVisAdaptiveIndIllum	×/×	Die Beschleunigungs-Technik B wird deaktiviert.
fgNoExploitLinkVisDirIllum	×/×	Die Beschleunigungs-Technik C wird beim Einsammeln von direkter Beleuchtung deaktiviert.
fgNoExploitLinkVisIndIllum	×/×	dito für indirekte Beleuchtung

Fortsetzung auf nächster Seite

Fortsetzung von voriger Seite		
Option	WB/WL	Beschreibung
fgIllumType <i>n</i> [1]	×/×	Beleuchtungstyp: 0≡kein Einsammeln, 1≡Einsammeln von direkter und indirekter Beleuchtung, 2≡Einsammeln von direkter Beleuchtung, 3≡Einsammeln von indirekter Beleuchtung
Sonstige		
scale <i>r</i> [50.0]	×/×	Faktor für die lineare Skalierung der berechneten Radiance-Werte zu Farbwerten, die sich für die Anzeige eignen. Der Wert -1.0 steht für eine automatische Skalierung. Im Radiance-Fall sollte die automatische Skalierung jedoch vermieden werden, da der Skalierungsfaktor andernfalls i. a. von einem bestimmten Augpunkt abhängig wäre.
gamma <i>r</i> [1.9]	×/×	Faktor für die Gamma-Korrektur
qualityCurvedObjects <i>n</i> [0]	×/×	Anfängliche Approximations-Qualität aller gekrümmten Objekte
noAdjust	×/×	Die Adjustierung von neuen Knoten der topologischen Datenstruktur auf die Objekt-Oberfläche wird deaktiviert.
noFacingTest	×/×	Der Facing-Test stellt sicher, daß zwischen zwei Patches höchstens dann ein Link eingerichtet wird, wenn sie einander zugewandt sind. Der Facing-Test unterdrückt Links, deren Transportkoeffizienten ausschließlich den Wert 0 haben.
standardFacingTest	×/×	Der Standard-Facing-Test wird anstelle einer etwas ausgefeilteren Variante verwendet. Der Standard-Facing-Test ist etwas effizienter. Allerdings erkennt er nicht alle Situationen, in denen zwei Patches einander zugewandt sind, da sich dieser Test nur auf die beiden Patch-Zentren bezieht.
nVisRaysInitLinking <i>n</i> [16]	×/×	Zwischen einem Paar von Ausgangs-Patches wird höchstens dann ein Link eingerichtet, wenn die Patches gegenseitig wenigstens teilweise sichtbar sind. Der Wert 0 deaktiviert diesen Sichtbarkeitstest.
noOccluderCaching	×/×	Das Ausnutzen von Strahlkohärenz durch einen Blockierer-Cache wird deaktiviert. Das Blockierer-Caching wirkt sich auf die Sichtbarkeitsberechnungen bei der Berechnung von Transportkoeffizienten und auf die Sichtbarkeits-tests beim initial linking aus.
multipleViewDirections	-/×	Bei der Extraktion einer sichtabhängigen Lösung werden mehrere Sichtrichtungen pro Ausgangs-Patch berücksichtigt. Vgl. Seite 59. Diese Option wird auch beim Final-Gathering berücksichtigt.
rotatePatchCSs <i>r</i> [0.0]	-/×	Alle Patch-Koordinatensysteme werden um ihre v -Achse entgegen dem Uhrzeigersinn gemäß diesem Winkel gedreht. Durch eine geeignete Wahl dieses Parameters können z. B. visuell störende Wechsel des gerichteten Bereichs in der Umgebung eines interessanten Augpunktes vermieden werden. Die Winkleinheit ist Grad.
noExtraHandlingDiffuse	-/×	Die Sonderbehandlung von diffusen Reflektoren wird deaktiviert.
writeTCs	×/×	Die Berechnung der Transportkoeffizienten und die adaptive numerische Integration werden in einer Datei mit Endung <code>.tcs</code> protokolliert.

Literaturverzeichnis

- [AH93] L. Aupperle and Pat Hanrahan. A Hierarchical Illumination Algorithm for Surfaces with Glossy Reflection. In *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 155–162, 1993. [14](#)
- [Arv93] James Arvo. Linear Operators and Integral Equations in Global Illumination. In *ACM SIGGRAPH '93 Course Notes - Global Illumination*, chapter 2, pages 1–21. 1993. [15](#), [18](#)
- [Aup93] Larry Aupperle. *Hierarchical Algorithms for Illumination*. Ph.D. thesis, Technical Report, Princeton, NJ, 1993.
- [BCR91] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transforms and numerical algorithms I. *Communications on Pure and Applied Mathematics*, 44:141–183, 1991. [28](#)
- [Ben97] Heinzgerd Bendels. Eine topologische Datenstruktur und ihre Anwendung im 3D-Graphiksystem MRT. Master's thesis, Universität Bonn, Institut für Informatik III, Bonn, Germany, 1997. [41](#)
- [Bey93] G. Beylkin. Wavelets and fast numerical algorithms. *Lecture notes for short course*, 1993.
- [BFS96] Heinzgerd Bendels, Dieter W. Fellner, and Stephan Schäfer. Hierarchical radiosity on topological data structures. In *B. Girod, H. Niemann, H.-P. Seidel, editors, 3D Image Analysis and Synthesis '96*, pages 111–118, 1996. [41](#)
- [Chr95] Per Henrik Christensen. *Hierarchical Techniques for Glossy Global Illumination*. Ph.D. thesis, Technical Report, Seattle, Washington, 1995. [7](#), [7](#), [18](#), [28](#), [30](#), [67](#)
- [CLSS97] Per H. Christensen, Dani Lischinski, Eric J. Stollnitz, and David H. Salesin. Clustering for glossy global illumination. *ACM Transactions on Graphics*, 16(1):3–33, January 1997.
- [CSSD94] Per Henrik Christensen, Eric J. Stollnitz, David H. Salesin, and Tony D. DeRose. Wavelet Radiance. In *Fifth Eurographics Workshop on Rendering*, pages 287–302, Darmstadt, Germany, June 1994.
- [CW93] Michael F. Cohen and John R. Wallace. *Radiosity and Realistic Image Synthesis*. Academic Press Professional, Boston, MA, 1993. [9](#)
- [Dau88] Ingrid Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41:909–996, October 1988. [20](#)
- [Fel96] Dieter W. Fellner. MRT - a teaching and research platform for 3d image synthesis. *IEEE CG&A*, 16(3), May 1996. [40](#)

- [Gla95] Andrew S. Glassner. *Principles of Digital Image Synthesis*. Morgan Kaufmann, San Francisco, CA, 1995. [9](#), [45](#)
- [Gor95] Steven J. Gortler. *Wavelet Methods for Computer Graphics*. Ph.D. thesis, Technical Report, Princeton, NJ, January 1995. [25](#)
- [GSCH93] Steven J. Gortler, Peter Schroder, Michael F. Cohen, and Pat Hanrahan. Wavelet Radiosity. In *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 221–230, 1993. [28](#)
- [GTGB84] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modelling the Interaction of Light Between Diffuse Surfaces. In *Computer Graphics (ACM SIGGRAPH '84 Proceedings)*, volume 18, pages 212–222, July 1984. [20](#)
- [Hec93] Paul S. Heckbert. Finite Element Methods for Radiosity. In *ACM SIGGRAPH '93 Course Notes - Global Illumination*, chapter 5, pages 1–7. 1993.
- [HSA91] Pat Hanrahan, David Salzman, and Larry Aupperle. A Rapid Hierarchical Radiosity Algorithm. In *Computer Graphics (ACM SIGGRAPH '91 Proceedings)*, volume 25, pages 197–206, July 1991. [7](#), [35](#), [37](#), [39](#)
- [ICG86] Dave S. Immel, Michael Cohen, and Donald P. Greenberg. A Radiosity Method for Non-Diffuse Environments. In *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, volume 20, pages 133–142, August 1986. [14](#), [20](#)
- [Kaj86] James T. Kajiya. The Rendering Equation. In *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, volume 20, pages 143–150, August 1986. [14](#)
- [Laf96] Eric Lafortune. *Mathematical Models and Monte Carlo Algorithms for Physically Based Rendering*. Ph.D. thesis, Leuven, Belgium, February 1996. [11](#)
- [LTG93] Daniel Lischinski, Filippo Tampieri, and Donald P. Greenberg. Combining Hierarchical Radiosity and Discontinuity Meshing. In *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 199–208, 1993. [44](#)
- [LW94] Eric P. Lafortune and Yves D. Willems. Using the Modified Phong BRDF for Physically Based Rendering. Technical Report CW197, Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium, November 1994. [12](#), [57](#)
- [Mal89] Stephane Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:674–693, July 1989. [20](#)
- [Mül97] Gordon Müller. Beschleunigung strahlbasierter Rendering-Algorithmen. Master's thesis, Universität Bonn, Institut für Informatik III, Bonn, Germany, 1997. [41](#), [41](#), [47](#)
- [PB94] Sumanta N. Pattanaik and Kadi Bouatouch. Haar Wavelet: A Solution to Global Illumination with General Surface Properties. In *Fifth Eurographics Workshop on Rendering*, pages 273–286, Darmstadt, Germany, June 1994.
- [Rei92] Mark C. Reichert. A Two-Pass Radiosity Method Driven by Lights and Viewer Position. M.Sc. thesis, Ithaca, NY, January 1992. [7](#), [43](#)

- [SAG94] Brian Smits, James Arvo, and Donald Greenberg. A Clustering Algorithm for Radiosity in Complex Environments. In *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 435–442, 1994. [7](#), [67](#)
- [SAS92] Brian E. Smits, James R. Arvo, and David H. Salesin. An Importance-Driven Radiosity Algorithm. In *Computer Graphics (ACM SIGGRAPH '92 Proceedings)*, volume 26, pages 273–282, July 1992. [67](#)
- [SAWG91] Francois Sillion, James R. Arvo, Stephen H. Westin, and Donald P. Greenberg. A Global Illumination Solution for General Reflectance Distributions. In *Computer Graphics (ACM SIGGRAPH '91 Proceedings)*, volume 25, pages 187–196, July 1991.
- [Sch94a] Christophe Schlick. A Survey of Shading and Reflectance Models. *Computer Graphics Forum*, 13(2):121–131, June 1994. [57](#)
- [Sch94b] Peter Schroder. *Wavelet Algorithms for Illumination Computations*. Ph.D. thesis, Technical Report, Princeton, NJ, November 1994. [14](#), [23](#), [28](#), [28](#), [30](#), [60](#)
- [Sch96a] Annette Scheel. Physikalisch basierte Beleuchtungsrechnung mit Wavelet-Radiance. Master's thesis, Universität Erlangen-Nürnberg, Technische Fakultät, Erlangen-Nürnberg, Germany, 1996. [57](#), [66](#), [67](#)
- [Sch96b] Peter Schroder. Wavelet Radiosity: Wavelet Methods for Integral Equations. In *ACM SIGGRAPH '96 Course Notes - Wavelets in Computer Graphics*, pages 143–165. 1996. [29](#)
- [Sch97] Stephan Schäfer. Hierarchical radiosity on curved surfaces. In Julie Dorsey and Phillip Slusallek, editors, *Rendering Techniques '97 (Proceedings of the Eighth Eurographics Workshop on Rendering)*, pages 187–192, New York, NY, 1997. Springer Wien. ISBN 3-211-83001-4. [43](#)
- [SDS95] Eric J. Stollnitz, Tony D. DeRose, and David H. Salesin. Wavelets for Computer Graphics: A primer part 1. *IEEE Computer Graphics and Applications*, 15(3):76–84, May 1995. [21](#)
- [SDS96] Eric J. Stollnitz, Tony D. DeRose, and David H. Salesin. *Wavelets for Computer Graphics: Theory and Applications*. Morgan Kaufmann, San Francisco, CA, 1996.
- [SGCH93] Peter Schroder, Steven J. Gortler, Michael F. Cohen, and Pat Hanrahan. Wavelet Projections for Radiosity. In *Fourth Eurographics Workshop on Rendering*, number Series EG 93 RW, pages 105–114, Paris, France, June 1993. [39](#)
- [Shi90] Peter Shirley. *Physically Based Lighting Calculations for Computer Graphics*. Ph.D. thesis, November 1990. [34](#)
- [SP94] Francois Sillion and Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann, San Francisco, CA, 1994. [9](#)
- [SSG⁺99] M. Stamminger, A. Scheel, X. Granier, F. Perez-Carzola, and G. Drettakis. Efficient glossy global illumination with interactive viewing. In *Proc. Graphics Interface '99*, 1999. to appear. [7](#)

- [SSS97a] H.-P. Seidel, P. Slusallek, and M. Stamminger. Global illumination computations with hierarchical finite element methods. In *Proc. Mathematical Methods for Curves and Surfaces '97*, 1997. Available from <http://www9.informatik.uni-erlangen.de/eng/research/pub1997>.
- [SSS97b] M. Stamminger, P. Slusallek, and H.-P. Seidel. Decoupling transport and reflection for radiance computations. In Stanislav Klimenko, editor, *Graphicon '97*, pages 51–58, Moscow, Russia, 1997. Available from <http://www9.informatik.uni-erlangen.de/eng/research/tr96>.
- [Sto83] Josef Stoer. *Einführung in die numerische Mathematik I*. Springer Verlag, Berlin, Heidelberg, New York, 1983. 34
- [Str91] Bjarne Stroustrup. *The C++ Programming Language*. 2nd ed., Addison-Wesley, Reading, Massachusetts, 1991. 40
- [War92] Gregory J. Ward. Measuring and Modeling Anisotropic Reflection. In *Computer Graphics (ACM SIGGRAPH '92 Proceedings)*, volume 26, pages 265–272, July 1992. 57
- [WH97] Andrew J. Willmott and Paul S. Heckbert. An empirical comparison of radiosity algorithms. Technical Report CMU-CS-97-115, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, April 1997. Available from <http://www.cs.cmu.edu/radiosity/emprad-tr.html>. 35
- [Zim98] Kurt Zimmermann. Developing the rendering equations. In *ACM SIGGRAPH '98 Course Notes 5 - A basic guide to Global Illumination*, chapter 6, pages 51–61. 1998.