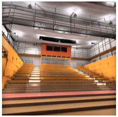
Kohärentes Metropolis-Lichttransportmodell mit Mutationen mit mehreren Versuchen

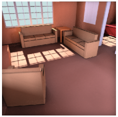
im Seminar Computergrafik

Jan Niklas Grieb

Georg-August-Universität Göttingen

11. Januar 2011





Inhalt

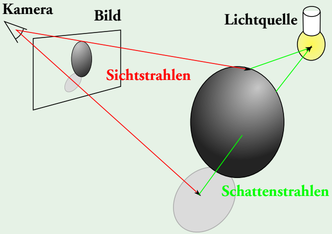
Beleuchtung in der Computergrafik

- Verfahren der Computergrafik:
  - Rasterizing: Grafikpipeline, hardwarebeschleunigt, in Echtzeit.
  - Strahlverfolgung: realistisch, indirekte Beleuchtung.
- Beleuchtung in der Szene:
  - Lokale Beleuchtung: Shading, Farbe der Oberfläche aus Blickrichtung, Licht und Material.
  - Bei der globalen Beleuchtung besteht der Anspruch, die Lichtausbreitung realistisch zu verfolgen (Geometrie, Physik, etc.).
- Fazit: Rasterizering/Shading ist schnell, Strahlverfolgung/globale Beleuchtung ist realistischer und rechenzeitintensiver.

Strahlverfolgung

- Beim Raycasting werden für jeden Sichtstrahl eines jeden Kamerapixels die Schnittpunkte mit der Szene bestimmt.
- Schnell, aber wenig Gestaltungsspielraum.
- Keine diffuse Beleuchtung, Schatten: nur Ja/Nein.

Illustration



Strahlverfolgung

- Vielzahl an Verbesserungen entwickelt:

Whitted-Raytracing

- Das Raytracing nach Turner Whitted berechnet auch Sekundärstrahlen rekursiv (Brechung, Spiegelung).
- Schatten (abrupte Kanten), aber noch keine diffusen Oberflächen implementiert.

Distribution Ray Tracing

- Hier werden mehrere Primär- und Sekundärstrahlen versendet (stochastisch verteilt) und gemittelt.
- Ergebnis: weichere Schattenverläufe, diffuse Beleuchtung.
- Realitätseindruck durch *motion blur* und Unschärfe gesteigert.

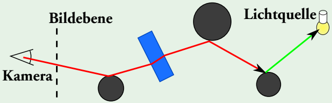
Pfadtracing

- Lösen der Rendergleichung von Kajiya bisher nicht gegeben: "Lichtausbreitung um die Ecke"

Abhilfe: Pfadtracing

- Das Pfadtracing generiert Sekundärstrahlen bei diffusen, reflektierenden und lichtdurchlässigen Materialien.
- Integration über mehrere Pfade pro Bildpixel.
- Mathematische Formulierung: Bidirektionale Reflektanz-, resp. Transmittanzverteilungsfunktion (BRDF, BTDF, etc.).

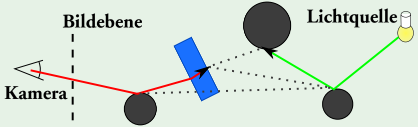
Illustration



Bidirektionales Pfadtracing I

- Verbesserung durch bidirektionalen Ansatz: gleichzeitige, unabhängige Konstruktion von Kamerastrahl und Lichtstrahl.

Illustration

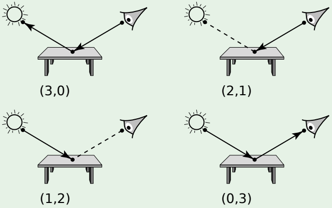


Bidirektionales Pfadtracing II

- (s, t): Kombination von Kamerapfaden der Länge s mit Lichtpfaden der Länge t zu Gesamtpfaden der Länge $k = s + t - 1$.

Illustration

- Alle (s, t)-Pfade der Länge $k = 2$.



[Quellenangaben zu den Bildern in der Ausarbeitung.]

Wahrscheinlichkeitsdichtefunktion und Erwartungswert

- Der Ewartungswert, wenn *Wahrscheinlichkeitsdichtefunktion* (PDF) verwendet:

$$E[f] = \int f(\mathbf{x})p(\mathbf{x})\,\mathrm{d}V. \tag{1}$$

- Sei F_N Schätzer aus N Samples für eine Größe I.
- Schätzer ist erwartungstreu für $E[F_N] = I$ und
- konsistent, falls $\lim_{N\to\infty} E[F_N - I] = 0$.

Monte-Carlo-Integration

- Verfahren zur numerischen Integration von Funktionen.

Standard-MC-Integration

$$I = \int_M f(\mathbf{x})\,\mathrm{d}\mu(\mathbf{x}), \quad F_N = \frac{1}{N}\sum_{i=1}^{N}\frac{f(\mathbf{X}_i)}{p(\mathbf{X}_i)} \quad \Rightarrow \quad E[F_N] = I, \tag{2}$$

wobei $\mathbf{X}_i$ Zufallszahlen nach PDF $p(\mathbf{x})$ aus Gebiet M mit Maß $\mathrm{d}\mu(\mathbf{x})$.

- Umgeht Fluch der Dimensionalität der gewöhnlichen Quadraturmethoden (Gauß, etc.), da Fehler $\delta I \propto 1/\sqrt{N}$, dimensionsunabhängig!

Metropolis-Hastings-Algorithmus I

- Bei der Verwendung von $p(\mathbf{x}) \propto f(\mathbf{x})$ werden die Zufallszahlen die "wichtigen" Stellen der Funktion treffen.
- Dies minimiert die Varianz von F_N und damit den Fehler der Monte-Carlo-Integration.
- Frage: Wie wird eine solche PDF $p(\mathbf{x})$ generiert?
- Antwort: Markov-Ketten!

Markov-Ketten

- Kette aus Realisierungen einer Zufallsvariablen
  $\ldots \to \mathbf{x}_i \to \mathbf{x}_{i+1} \to \mathbf{x}_{i+2} \to \ldots$
- Maß für die Übergangswahrscheinlichkeit ist $K(\mathbf{x}_i \to \mathbf{x}_{i+1})$ (keine Vorgeschichte).
- Wahrscheinlichkeit für Zustand $\mathbf{x}$ in Schritt i: $P(\mathbf{x}, i)$.
- stationär: $P(\mathbf{x}, i) = P(\mathbf{x}, i+1) \equiv P(\mathbf{x})$

Detailliertes Gleichgewicht

- Fordere

$$p(\mathbf{x}_i)K(\mathbf{x}_i \to \mathbf{x}_j) = p(\mathbf{x}_j)K(\mathbf{x}_j \to \mathbf{x}_i). \tag{3}$$

- Dann: $P(\mathbf{x}) \propto p(\mathbf{x})$

Metropolis-Hastings-Algorithmus II

- Implementation einer Markov-Kette, welche $p(\mathbf{x})$ reproduziert
- durch *Vorschlagen* von Kandidaten mit Wahrscheinlichkeit $T(\mathbf{x}_i \to \mathbf{x}_j)$ und
- *Akzeptanz* mit Wahrscheinlichkeit

$$W(\mathbf{x}_i \to \mathbf{x}_j) = \min\left(1, \frac{p(\mathbf{x}_j)T(\mathbf{x}_j \to \mathbf{x}_i)}{p(\mathbf{x}_i)T(\mathbf{x}_i \to \mathbf{x}_j)}\right) \tag{4}$$

  (so groß wie möglich).
- $\Rightarrow K(\mathbf{x}_i \to \mathbf{x}_j) = T(\mathbf{x}_i \to \mathbf{x}_j)W(\mathbf{x}_i \to \mathbf{x}_j)$
- Wichtig: $p(\mathbf{x})$ muss i.A. nicht berechenbar sein, es reicht die Verhältnisse zu kennen.

Metropolis-Hastings-Algorithmus III

Metropolis-Hastings-Algorithmus

1) Startwert $\mathbf{x}_i \leftarrow \mathbf{x}_0$
2) Vorschlag $\mathbf{x}^*$ gemäß $T(\mathbf{x}_i \rightarrow \mathbf{x}^*)$
3) Berechne $w \equiv W(\mathbf{x}_i \rightarrow \mathbf{x}^*) = [p(\mathbf{x}^*)T(\mathbf{x}^* \rightarrow \mathbf{x}_i)]/[p(\mathbf{x}_i)T(\mathbf{x}_i \rightarrow \mathbf{x}^*)]$.
   i) Wenn $rand(0,1) \leq w$, so akzeptiere Vorschlag: $\mathbf{x}_{i+1} \leftarrow \mathbf{x}^*$.
   ii) Ansonsten alter Wert: $\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i$.
4) Sample $\mathbf{x}_{i+1}$ in Histogramm
5) Setze $\mathbf{x}_i \leftarrow \mathbf{x}_{i+1}$ und beginne erneut bei 2).

Startup-Bias

Die Markov-Kette ist zunächst vom Startwert abhängig und somit erst nach einiger Zeit stochastisch unabhängig.

MC-Integration in der CG

- In einem Pfadtracer die Rendergleichung durch MC-Integration lösen:

$$I = \int_{\Omega} f(\bar{\mathbf{x}}) \mathrm{d}\mu(\bar{\mathbf{x}}). \tag{5}$$

Pfadintegral-Formulismus

- Erweitere Messgleichung rekursiv mit Rendergleichung:

$$\begin{aligned} I = & \int_{\mathcal{M}^2} L_e(\mathbf{x}_1 \to \mathbf{x}_0) G(\mathbf{x}_1 \leftrightarrow \mathbf{x}_0) W_e(\mathbf{x}_1 \to \mathbf{x}_0) \, \mathrm{d}A(\mathbf{x}_0) \, \mathrm{d}A(\mathbf{x}_1) \\ & + \int_{\mathcal{M}^3} L_e(\mathbf{x}_2 \to \mathbf{x}_1) G(\mathbf{x}_2 \leftrightarrow \mathbf{x}_1) f_s(\mathbf{x}_2 \to \mathbf{x}_1 \to \mathbf{x}_0) \\ & \quad \cdot G(\mathbf{x}_2 \leftrightarrow \mathbf{x}_1) W_e(\mathbf{x}_1 \to \mathbf{x}_0) \, \mathrm{d}A(\mathbf{x}_0) \, \mathrm{d}A(\mathbf{x}_1) \, \mathrm{d}A(\mathbf{x}_2) + \dots \end{aligned} \tag{6}$$

Pfadintegrale I

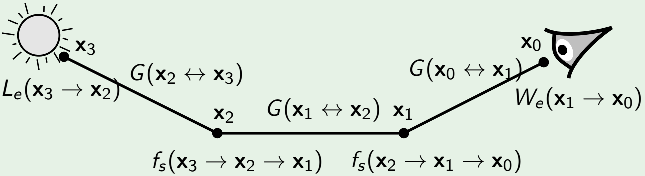
- Beim PI-Formulismus wird die unabhängige Summe über die Integration aller Pfade einer Länge k gebildet.
- Diese Integrale werden mit Monte-Carlo-Verfahren gesampelt.
- Beitrag $f(\bar{\mathbf{x}})$ eines Pfades $\bar{\mathbf{x}}$ zum MC-Schätzwert des Integrals: vertexweise aus BRDF f_s und geometrischem Faktor G multipliziert.

Pfadintegrale II

- Hier ein Beispiel für $k = 3$ ($\bar{\mathbf{x}} = \mathbf{x}_0 \dots \mathbf{x}_3$):

$$f(\bar{\mathbf{x}}) = L_e(\mathbf{x}_0 \to \mathbf{x}_1) G(\mathbf{x}_0 \leftrightarrow \mathbf{x}_1) f_s(\mathbf{x}_0 \to \mathbf{x}_1 \to \mathbf{x}_2) G(\mathbf{x}_1 \leftrightarrow \mathbf{x}_2) \\ \cdot f_s(\mathbf{x}_1 \to \mathbf{x}_2 \to \mathbf{x}_3) G(\mathbf{x}_2 \leftrightarrow \mathbf{x}_3) W_e(\mathbf{x}_2 \to \mathbf{x}_3)$$

Illustration



Pfadwahrscheinlichkeit

- Die Pfadwahrscheinlichkeit $p(\mathbf{x})$ ergibt sich aus den Wahrscheinlichkeiten pro Flächenelement an den Vertizes:

$$p(\bar{\mathbf{x}}) = \frac{\mathrm{d}P}{\mathrm{d}\mu}(\bar{\mathbf{x}}) = \prod_{i=0}^{k} \frac{\mathrm{d}P}{\mathrm{d}A}(\mathbf{x}_i) \tag{7}$$

- Bei Lichtquellen ist Wahrscheinlichkeit pro Flächenelement $\mathrm{d}A$ *a-priori* vorgegeben.
- Bei Sekundärstrahlen wird Richtung ω_0 mit W'keit $p(\omega_0)$ gewählt. Transformation der PDF:

$$p(\mathbf{x}') = p(\omega_0)\frac{|\cos(\theta_i')|}{\|\mathbf{x} - \mathbf{x}'\|^2}$$

Fazit

- Die allgemeine Monte-Carlo-Integration in Pfadtracern mittels PI kann globale Beleuchtung realisieren, hat aber Schwächen bei schwierigen Beleuchtungsverhältnissen.
- Manche Effekte (e. g. spekulare Lichtquellen) nur mit BPT realisierbar

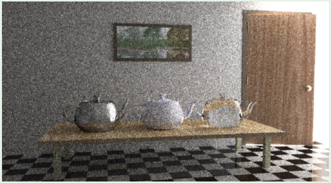
Multiple-Importance-Sampling

- Die einzelnen Pfade kommen bei unterschiedlichen Importance-Sampling-Techniken zustande.
- Optimales Endergebnis durch gewichtetes Mittel aus allen Techniken: *multi-sample model*.
- Unverzichtbar bei BPT.

Metropolis-Lichttransportmodell

- Beim Metropolis-Light-Transport (MLT) wird auf Metropolis-Hastings (MH) zurückgegriffen, so dass die intensiven Lichtpfade bevorzugt behandelt werden. Literatur: [Veach 1997], [Veach and Guibas 1997]

Vergleich bidir. Pathtracer – MLT



- Bessere Konvergenz bei schlechten Lichtverhältnissen (e. g. Türspalt).

Metropolis-Lichttransportmodell

- Verwende MH zum Samplen von Lichtpfaden:

MLT-Algorithmus

1) Startpfad $\overline{\mathbf{x}}_i \leftarrow \overline{\mathbf{x}}_0$
2) Vorschlag $\overline{\mathbf{x}}^*$ gemäß $T(\overline{\mathbf{x}}_i \rightarrow \overline{\mathbf{x}}^*)$
3) Berechne $w = W(\overline{\mathbf{x}}_i \rightarrow \overline{\mathbf{x}}^*) = [p(\overline{\mathbf{x}}^*)T(\overline{\mathbf{x}}^* \rightarrow \overline{\mathbf{x}}_i)]/[p(\overline{\mathbf{x}}_i)T(\overline{\mathbf{x}}_i \rightarrow \overline{\mathbf{x}}^*)]$.
   i) Für $rand(0,1) \geq 1$ akzeptiere den Vorschlag: $\overline{\mathbf{x}}_{i+1} \leftarrow \overline{\mathbf{x}}^*$.
   ii) Ansonsten behalte den alten Wert: $\overline{\mathbf{x}}_{i+1} \leftarrow \overline{\mathbf{x}}_i$.
4) Lichtintensität von $\overline{\mathbf{x}}_{i+1}$ in Histogramm
5) Setze $\overline{\mathbf{x}}_i \leftarrow \overline{\mathbf{x}}_{i+1}$ und beginne erneut bei 2) bis n MH-Mutationen durchgeführt sind.

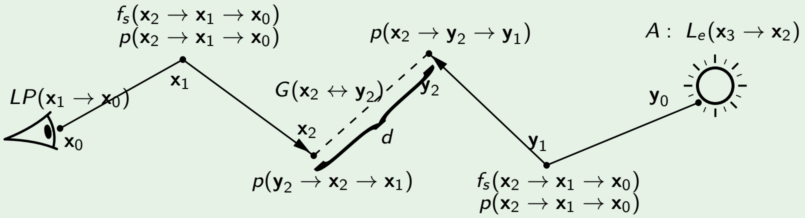
- Bei Farben wird die Lichtintensität auf die photometrische *Luminanz* skaliert.

Lichtpfadintensität

- Bidirektionales MLT: Kombiniere Kamera- und Lichteilpfad zu (s,t)-Pfaden der Gesamtlänge $k = s + t - 1$.

$$\frac{f_s(\mathbf{x}_2 \to \mathbf{x}_1 \to \mathbf{x}_0)}{p(\mathbf{x}_2 \to \mathbf{x}_1 \to \mathbf{x}_0)} \times \frac{G(\mathbf{x}_2 \leftrightarrow \mathbf{y}_2) f_s(\mathbf{y}_2 \to \mathbf{x}_2 \to \mathbf{x}_1) f_s(\mathbf{x}_2 \to \mathbf{y}_2 \to \mathbf{y}_1)}{\pi d^2}$$
$$\times \frac{f_s(\mathbf{y}_2 \to \mathbf{y}_1 \to \mathbf{y}_0)}{p(\mathbf{y}_2 \to \mathbf{y}_1 \to \mathbf{y}_0)} \times \frac{A L_e(\mathbf{y}_0 \to \mathbf{y}_1)}{P_{\mathrm{len}} P_{\mathrm{light}}} = LP(\mathbf{x}_1 \to \mathbf{x}_0)$$

Beispielpfad im bidirektionalen MLT



MLT-Mutationsstrategien

- Gestalte Übergänge zu Mutationen $T(\mathbf{x}_i, \mathbf{x}_j)$ so, dass dieses berechenbar ist und Ergodizität vorliegt.
- Ergodizität wird in MLT durch komplett neue Teilpfade realisiert (*birectional mutations*).
- Pfadmutationen an den Vertizes werden nach komplizierten Mustern je nach Oberfläche / Material behandelt:
  - lens perturbations,
  - caustic perturbations,
  - multi-chain perturbations
- Kamerateilpfadmutationen, damit jeder Kamerapixel gleich oft vorgeschlagen (*stratification* abbauen).

Normierung des Histogramms

- Normiere das Histogramm:

$$I_{\mathrm{norm}} = sI, \quad s = \frac{f_{\mathrm{avg}}}{h_{\mathrm{avg}}}. \tag{8}$$

f_{avg} mittlere Intensität
h_{avg} mittlere Anzahl von Samples pro Bin

- Ersteres aus gewöhnlichem BPT mit MIS,
- dann Rauschunterdrückung und Abgrasen von Kaustiken mittels MLT.

Verbesserungen an MLT?

Weiterführende Arbeiten

- [Kelemen et al. 2002] verbessern MLT durch Beschreibung von *kleinen Mutationen* durch Zufallszahlen auf Einheitswürfel (Varianz verringert). *Große Mutationen* für Ergodizität.
- Verwende MH in Energieverteilungs-Pfadtracern [Cline et al. 2005].
- Oder *Replica Exchange* ersetzt MH [Kitaoka et al. 2009], erinnert an *simulated annealing*.

- Die sequenzielle Berechnung der Mutationen ist ein großer Schwachpunkt von MLT: keine Parallelisierung.
- Zudem gering Cache-Freundlichkeit
- und Flickerprobleme in Animationsszenarien.

Inhalt

Multiple-Try-Mutationen

- In vielen Dimension wächst die Schrittgröße (multivariate Normalverteilungen) mit $\sqrt{d}$: curse of dimensionality. Dies führt zu vielen Rückweisungen.
- Bei verringerter Schrittweite wird der Raum weniger gut abgedeckt.

Multiple-Try-Metropolis-Hastings

- [Liu et al. 2000] haben eine Verbesserung von MH vorgeschlagen, bei der mehrere Kandidaten gegeneinander antreten (*multiple-try*).

Multiple-Try-Metropolis-Hastings

- Die Vorschlagswahrscheinlichkeitsverteilung sollte

$$T(\mathbf{x} \to \mathbf{y}) > 0 \Leftrightarrow T(\mathbf{y} \to \mathbf{x}) > 0 \tag{9}$$

  erfüllen. Mit symmetrischer, nicht-negativer Funktion $\lambda(\mathbf{x}, \mathbf{y})$ definiert man:

$$w(\mathbf{x}, \mathbf{y}) := p(\mathbf{y})\, T(\mathbf{y} \to \mathbf{x}) \lambda(\mathbf{x}, \mathbf{y}). \tag{10}$$

- Diese Funktion vergleicht mehrere Kandidaten / Widersacher miteinander. Im Falle von detailliertem Gleichgewicht realisiert die resultierende Markov-Kette die PDF $p(\mathbf{x})$.
- Möglichkeit:

$$\lambda(\mathbf{x}, \mathbf{y}) = [\, T(\mathbf{y} \to \mathbf{x})\, T(\mathbf{x} \to \mathbf{y})]^{-1} \tag{11}$$

Multiple-Try-Metropolis-Hastings

MTMH-Algorithmus

1) Startwert $\mathbf{x}_i \leftarrow \mathbf{x}_0$
2) Vorschlag: k neue $\{\mathbf{y}_l\}$ gemäß $T(\mathbf{x}_i \rightarrow \mathbf{y}_l)$
3) Wähle ein $\mathbf{x}^* = \mathbf{y}_{l^*}$ mit Wahrscheinlichkeit gemäß Gewicht $w(\mathbf{x}_i, \mathbf{y}_l)$ aus.
4) Ziehe $k-1$ Widersacher $\{\mathbf{z}_l\}$ gemäß $T(\mathbf{x}^* \rightarrow \mathbf{z}_l)$ und setze $\mathbf{z}_k := \mathbf{x}_i$.
5) Berechne $\tilde{w} := [\sum_l w(\mathbf{x}_i, \mathbf{y}_l)]/[\sum_l w(\mathbf{x}^*, \mathbf{z}_l)]$.
   i) Für $rand(0,1) \leq \tilde{w}$ akzeptiere den Vorschlag: $\mathbf{x}_{i+1} \leftarrow \mathbf{x}^*$.
   ii) Ansonsten behalte den alten Wert: $\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i$.
6) Sample $\mathbf{x}_{i+1}$ in Histogramm
7) Setze $\mathbf{x}_i \leftarrow \mathbf{x}_{i+1}$ und beginne erneut bei 2).

- Erfüllt detailliertes Gleichgewicht!

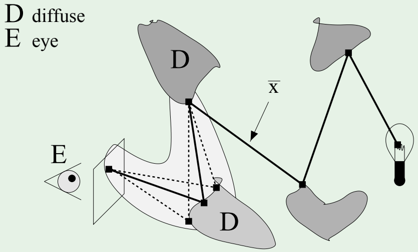
Multiple-Try-Metropolis-Hastings in MLT I

- Vektorielle Verabeitung von Pfadmutationen durch Verwendung von MTMH in MLT: kohärentes Metropolis-Lichttransportmodell (*coherent Metropolis Light Transport*, CMLT).
- Mehrere Kandidaten gleichzeitig berechnet und verglichen.
- effiziente Verschachtelung von "normalem" MH und MTMH: lichtintensive MH-Pfade werden dann mit MTMH behandelt (Kaustiken "absuchen").

Multiple-Try-Metropolis-Hastings in MLT II

- Mutation der Kamerateilpfade: $(L|D)S^*E$.

Illustration MTMH-Mutationen



Anpassung des Verfahrens

- CMLT beinhaltet MLT, wobei die Kameraunterpfade des bidirektionalen MC-Pfadtracers kohärent durch MTMH-Störungen gesampelt wird.
- Im Gegensatz zum ursprünglichen MTMH-Algorithmus von [Liu et al. 2000] werden alle Kandidaten ausgewertet und nach den Metropolis-Raten gewichtet gemittelt. Dies erfüllt immer noch das detaillierte Gleichgewicht.
- Die MTMH-Störungen sind (biasfreie) Linsen- und kaustische Mutationen.

Kohärenter Metropolis-Lichttransport

Äusserer Algorithmus

1) Startpfad $\overline{\mathbf{x}}_i \leftarrow \overline{\mathbf{x}}_0$

   A)-G) [...]

2) Vorschlag: $\overline{\mathbf{x}}^*$ gemäß $T_{\text{bidir}}(\overline{\mathbf{x}}_i \to \overline{\mathbf{x}}^*)$.
3) Berechne $w = [p(\overline{\mathbf{x}}^*)T_{\text{bidir}}(\overline{\mathbf{x}}_i \to \overline{\mathbf{x}}^*)]/[p(\overline{\mathbf{x}}_i)T_{\text{bidir}}(\overline{\mathbf{x}}^* \to \overline{\mathbf{x}}_i)]$.
   i) Für $rand(0,1) \leq w$ akzeptiere den Vorschlag: $\overline{\mathbf{x}}_{i+1} \leftarrow \overline{\mathbf{x}}^*$.
   ii) Ansonsten behalte den alten Wert: $\overline{\mathbf{x}}_{i+1} \leftarrow \overline{\mathbf{x}}_i$.
4) Setze $\overline{\mathbf{x}}_i \leftarrow \overline{\mathbf{x}}_{i+1}$ und beginne erneut bei 2) bis n MH-Mutationen durchgeführt sind.

Kohärenter Metropolis-Lichttransport

Innerer Algorithmus

A) Startpfad $\bar{\mathbf{x}}_i^{(j)} \leftarrow \bar{\mathbf{x}}_i^{(0)} \leftarrow \bar{\mathbf{x}}_i$

B) Vorschlag: k neue $\{\bar{\mathbf{y}}_l\}$ gemäß $T_{\mathrm{lc}}(\bar{\mathbf{x}}_i^{(j)} \rightarrow \bar{\mathbf{y}}_l)$

C) Wähle ein $\bar{\mathbf{x}}^* = \bar{\mathbf{y}}_{l^*}$ mit Wahrscheinlichkeit gemäß Gewicht $w(\bar{\mathbf{x}}_i^{(j)}, \bar{\mathbf{y}}_l)$ aus.

D) Ziehe $k-1$ Widersacher $\{\bar{\mathbf{z}}_l\}$ gemäß $T_{\mathrm{lc}}(\bar{\mathbf{x}}^* \rightarrow \bar{\mathbf{z}}_l)$ und setze $\bar{\mathbf{z}}_k := \bar{\mathbf{x}}_i^{(j)}$.

E) Berechne $\tilde{w} := [\sum_l w(\bar{\mathbf{x}}_i^{(j)}, \bar{\mathbf{y}}_l)]/[\sum_l w(\bar{\mathbf{x}}^*, \bar{\mathbf{z}}_l)]$.

   I) Für $rand(0,1) \leq \tilde{w}$ akzeptiere den Vorschlag: $\bar{\mathbf{x}}_i^{(j+1)} \leftarrow \bar{\mathbf{x}}^*$.

   II) Ansonsten behalte den alten Wert: $\bar{\mathbf{x}}_i^{(j+1)} \leftarrow \bar{\mathbf{x}}_i$.

F) $2k$ Lichtintensitäten mit Gewichten $w\frac{f(\bar{\mathbf{y}}_l)}{F_y}$ und $(1-w)\sum_{l=1}^{k}\frac{f(\bar{\mathbf{z}}_l)}{F_z}$, $l = 1\ldots k$, in Histogramm

G) Setze $\bar{\mathbf{x}}_i^{(j)} \leftarrow \bar{\mathbf{x}}_i^{(j+1)}$ und beginne erneut bei B) bis m MTMH-Mutationen durchgeführt sind.

- $F_y = \sum_{l=1}^{k} f(\bar{\mathbf{y}}_l)$, $F_z = \sum_{l=1}^{k} f(\bar{\mathbf{z}}_l)$, wieder Luminanz-skaliert.

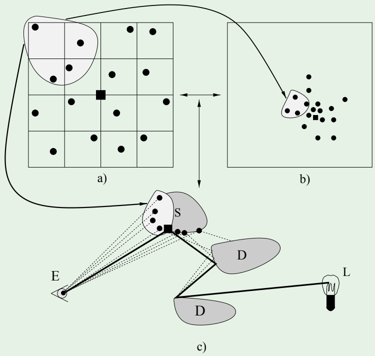
Parameterwahl

- [Segovia et al. 2007] finden keine Abhängigkeit von der Wahl von λ.
- Kritische Parameter sind die Varianzen der Gaußschen Mutationsparameter und die Anzahl k von MTMH-Vorschlägen.

Kohärenz

- Generierung von kohärenten Strahlbündeln: uniform verteilte, geclusterte Mutationsschritte. Dann: *Box-Muller-Trafo*.
- Vektorielle SIMD-Berechnung der Cluster.
- *Auch möglich*: Strahlbündel-Frustums [Reshevtov et al. 2005].

Illustration der kohärenten Bündel



Resultate I

- [Segovia et al. 2005] verwenden OpenRT.
- Parameterstudien für verschachteltes CMLT.

Parameterabhängigkeit I

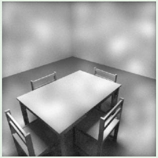
- Keine Sensitivität auf die Länge m der MTMH-Untersequenzen,
- da MH-Samples bereits proportional zur Lichtintensität sind.

Resultate II

Parameterabhängigkeit II

- Zusammenhang von Schrittgröße σ und Anzahl an MTMH-Wettstreitern.
- Viele Kandidaten kompensieren große Schrittweite und stabilisieren damit Algorithmus.
- Zu viele Kandidaten sorgen für *stratification*.

Beispiel für *stratification*



Resultate III

- Leistungssteigerug durch SSE-SIMD: Faktor 1.5 bis 2.5.
- Cachefreundlichkeit: Vergleich von CMLT mit MLT und Radiosity zeigt, dass die kohärente Arbeitsweise die Cachezugriffe auf Niveau des letzteren Verfahrens senkt.

Inhalt

Anwendung & Beschränkungen

- Anwendung in kommerziellen Renderern in Aussicht.
- Zudem Übertragung auf andere Bereiche möglich, wie e. g. *energy-redistribution* rendering.

Einschränkungen

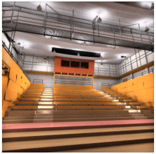
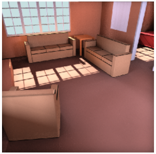
- Weiterhin Flickeringprobleme, evtl. Glättung der Skalierung und gespeicherte Seeds.
- Weiterhin Bedarf eines guten (Pseudo)-RNG.
- Rechnernetzwerke: kopiere Geometrie, Schreibzugriff auf alle Teile des Bildes.

Abschliessende Worte

- MLT stellt ein numerisch stabiles Rendering-Verfahren mit globaler Beleuchtung (geringe Ansprüche an Szene und Lichtverhältnisse) ohne Bias.
- CMLT verbessert die Implementation von MLT auf Parallelrechnern durch vektorielle Berechnung von Mutationen und Cache-Freundlichkeit.
- Guter Schritt in Richtung Echtzeit-Pfadtracing,
- Kelemen-MLT neuerdings auf GPU gerechnet.

Ende

- Vielen Dank für Ihre Aufmerksamkeit.


- Besten Dank auch an W. Kurth für die Betreuung des Seminars und an R. Hemmerling für seine Unterstützung und Hilfestellungen.

Inhalt

Verweise I

Cline, D., and Egbert, P. 2005.
A practical introduction to metropolis light transport.
Tech. rep., Brigham Young University, May.

Cline, D., Talbot, J., and Egbert, P. 2005.
Energy redistribution path tracing.
ACM Transactions on Graphics 24, 3 (Aug.), 1186–1195.

Heckbert, P. S. 1990.
Adaptive radiosity textures for bidirectional ray tracing.
In *Computer Graphics (Proceedings of SIGGRAPH 90)*, 145–154.

Kajiya, J. T. 1986.
The rendering equation.
In *Computer Graphics (Proceedings of SIGGRAPH 86)*, 143–150.

Kelemen, C., Szirmay-Kalos, L., Antal, G., and Csonka, F. 2002.
A simple and robust mutation strategy for the metropolis light transport algorithm.
Computer Graphics Forum 21, 3, 531–540.

Keller, A. 1997.
Instant radiosity.
In *Proceedings of SIGGRAPH 97*, Computer Graphics Proceedings, Annual Conference Series, 49–56.

Kitaoka, S., Kitamura, Y., and Kishino, F. 2009.
Replica exchange light transport.
Computer Graphics Forum 28, 8 (Dec.), 2330–2342.

Verweise II

Liu, J. S., Liang, F., and Wong, W. H. 2000.
The Multiple-Try Method and Local Optimization in Metropolis Sampling.
Journal of the American Statistical Association 95, 449 (March), 121+.

Pharr, M. 2003.
Chapter 9: Metropolis sampling.
In *SIGGRAPH 2003 Course Note #44: Monte Carlo Ray Tracing*.

Reshetov, A., Soupikov, A., and Hurley, J. 2005.
Multi-level ray tracing algorithm.
ACM Transactions on Graphics 24, 3 (Aug.), 1176–1185.

Segovia, B., Iehl, J.-C., and Péroche, B. 2007.
Coherent Metropolis Light Transport with Multiple-Try Mutations.
Tech. Rep. RR-LIRIS-2007-015, LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/Ecole Centrale de Lyon, Apr.

Veach, E., and Guibas, L. J. 1997.
Metropolis light transport.
In *Proceedings of SIGGRAPH 97*, Computer Graphics Proceedings, Annual Conference Series, 65–76.

Veach, E. 1997.
Robust Monte Carlo Methods for Light Transport Simulation.
PhD thesis, Stanford.

Wald, I., Slusallek, P., Benthin, C., and Wagner, M. 2001.
Interactive rendering with coherent ray tracing.
Computer Graphics Forum 20, 3, 153–164.

Wahrscheinlichkeitsdichtefunktion

- Eine Wahrscheinlichkeitsdichtefunktion (*probability density function*, PDF) $p(x)$ misst die infinitesimale Wahrscheinlichkeit einer Zufallsvariablen X:

$$P(a < X < b) = \int_a^b p(x)\,\mathrm{d}x. \tag{12}$$

- Meist wird eine PDF normiert: $\int_{-\infty}^{\infty} p(x)\,\mathrm{d}x = 1$.
- Die Stammfunktion wird (kumulative) Verteilungsfunktion genannt:

$$F(x) = \int_{-\infty}^{x} p(x')\,\mathrm{d}x'. \tag{13}$$

Lichtpfadnotation

- reguläre Ausdrücke in Lichtpfaden [Heckbert 1990]

  L Lichtquelle,
  D diffuse Oberfläche,
  S spiegelnde Oberfläche,
  E Beobachter.

Beispiel

$$ES^{*}DS^{+}(D|L)$$

kaustischer Lichtpfad von der Linse zu einer diffusen Oberfläche oder einer Lichtquelle

Verteilungsfunktionen

- BRDF (*bidirectional reflectance distribution function*)

Definition

$$f_s(x \to x' \to x'') = f_s(x \to x', x \to x'') = \frac{\mathrm{d}L_r(x \to x'')}{\mathrm{d}E_i(x \to x')} = \frac{\mathrm{d}L_r(x \to x'')}{L_i(x \to x')\cos\theta_0\,\mathrm{d}(x \to x')},$$

wobei E_i Lichteinfluss und L_r -ausfluss, sowie θ_0 der Winkel zwischen Einfallsvektor $x \to x'$ und der Oberflächen-Normalen sind.

- lichtdurchlässige Materialien: BTDF (*bidirectional transmittance distribution function*)
- Zusammen ergeben diese die BSDF (*birectional scattering distribution function*).

Rendergleichung

- Die Rendergleichung wurde von [Kajiya 1986] aufgestellt.

Rendergleichung

Notation von [Veach 1997]:

$$\begin{aligned} L(x' \to x'') = {} & L_e(x' \to x'') + \\ & \int_{\mathcal{M}} L(x \to x') f_s(x \to x' \to x'') G(x \leftrightarrow x') \, \mathrm{d}A(x) \end{aligned} \tag{14}$$

Rendergleichung

- Die Rendergleichung wurde von [Kajiya 1986] aufgestellt.

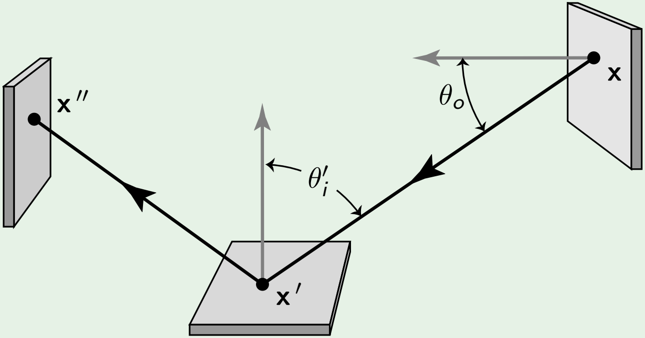
Rendergleichung

$L_{(e)}$	(emittierter) Licht-Energiefluss
G	geometrischer Faktor (Skalarprodukt)
V	visueller Faktor
f_s	*bidirectional scattering distribution function* (BSDF)
M	Vereinigung aller Oberflächen
dA	entsprechendes Lebesque-Maß

Rendergleichung

- Die Rendergleichung wurde von [Kajiya 1986] aufgestellt.

Illustration



Pfadintegralformulismus I

- Die Messgleichung eines hypothetischen Sensors, der mit Responsivität $W_e(\mathbf{x} \to \mathbf{x}')$ auf Licht $L(\mathbf{x} \to \mathbf{x}')$ reagiert:

$$I = \int_{\mathcal{M}\times\mathcal{M}} W_e(\mathbf{x} \to \mathbf{x}')L(\mathbf{x} \to \mathbf{x}')G(\mathbf{x} \leftrightarrow \mathbf{x})\,\mathrm{d}A(x)\,\mathrm{d}A(x).$$

- Erweitere rekursiv mit der Rendergleichung (s.o.), um die Pfadintegralgleichung zu erhalten:

$$I = \sum_{k=1}^{\infty} \int_{\mathcal{M}^{k+1}} \Bigg[L_e(\mathbf{x}_k \to \mathbf{x}_{k-1})G(\mathbf{x}_k \leftrightarrow \mathbf{x}_{k-1})W_e(\mathbf{x}_1 \to \mathbf{x}_0) \cdot \left(\prod_{i=1}^{k-1} f_s(\mathbf{x}_{i+1} \to \mathbf{x}_i \to \mathbf{x}_{i-1})G(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+1}) \right) \mathrm{d}A(\mathbf{x}_0)\ldots\mathrm{d}A(\mathbf{x}_k) \Bigg]$$

Pfadintegralformulismus II

- Beim Pfadintegral-Formulismus wird der Integrationsraum in die Teilräume der endlichen Pfadlängen k zerlegt:

$$\Omega = \bigcup_{k=1}^{\infty} \Omega_k, \quad \bar{\mathbf{x}} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k) \in \Omega_k$$

- Das Maß dieser Mengen wird kombiniert:

$$\mu(G) = \sum_{k=1}^{\infty} \mu_k(G \cap \Omega_k), \quad \mathrm{d}\mu_k(\bar{\mathbf{x}}) = \mathrm{d}A(\mathbf{x}_0) \ldots \mathrm{d}A(\mathbf{x}_k)$$

- Integral ist dann die Summe über Pfade aller Längen.
- Beitrag eines Pfades: multiplizierte BSDF und geometrischen Faktoren.

Umrechnung Raumwinkel

- $\omega_0 = \widehat{\mathbf{x}' - \mathbf{x}}$ Raumwinkel der Gerade $\mathbf{x} \to \mathbf{x}'$.
- Projizierter Raumwinkel von $D \subset S^2$:

$$\sigma_{\mathbf{x}}^{\perp}(D) = \int_D |\omega \cdot \mathbf{N}(\mathbf{x})| \, \mathrm{d}\sigma(\omega)$$

- infinitesimal:

$$\mathrm{d}\sigma^{\perp}(\omega_0) = |\omega_0 \cdot \mathbf{N}(\mathbf{x})| \, \mathrm{d}\sigma(\omega_0),$$

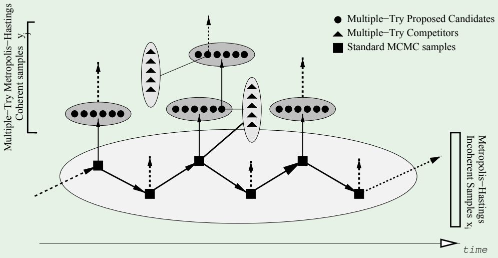
- $\mathbf{N}(\mathbf{x})$ Normalenvektor der Oberfläche an Stelle $\mathbf{x}$, so dass $\omega_0 \cdot \mathbf{N}(\mathbf{x}) = \cos\theta_0$.
- Umrechnung PDF:

$$p^{\perp}(\omega_0) = p(\omega_0) \frac{1}{\cos\theta_0}$$

Multiple-Try-Metropolis-Hastings in MLT II

- Illustration der MTMH-Wettstreiter (Kreise) und Widersacher (Dreiecke) als zweite Ebene unterhalb MLT-Mutationen (Quadrate).

Illustration MTMH-Mutationen: Zeit



Multiple-Try-Metropolis-Hastings in MLT II

- Illustration der MTMH-Wettstreiter (Kreise) und Widersacher (Dreiecke) als zweite Ebene unterhalb MLT-Mutationen (Quadrate).

Illustration MTMH-Mutationen: Screen