

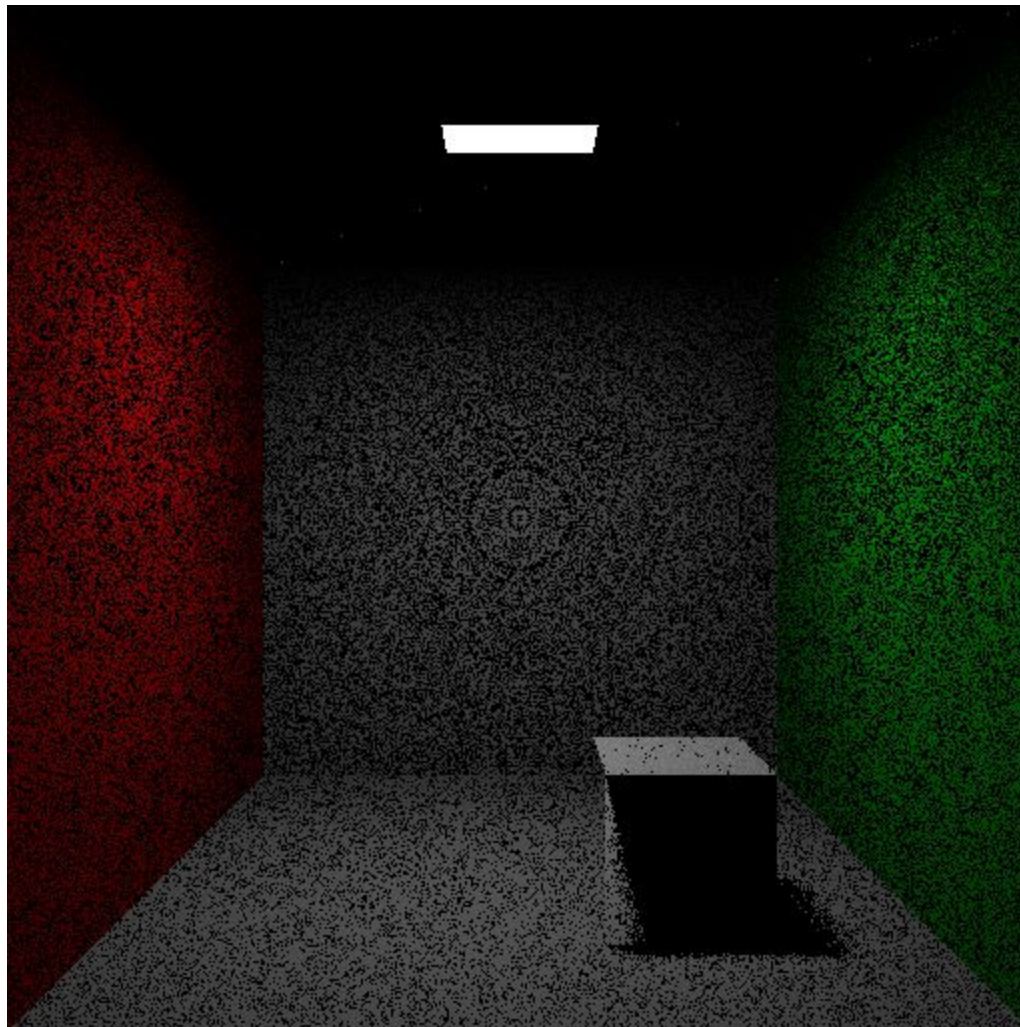
Multiple Importance Sampling

Philip Berndt
Michael Albrecht

64 Samples

512x512px

30min Time



Light Source Area Sampling

- Wiederhole für anzahl an samples:
- Algorithmus iteriert über alle Lightsources
- wenn zusammenhängende Lightsource gefunden wurde berechne gesamt Fläche
- erstelle Verteilung über alle Lightsources die zusammenhängen in abhängigkeit von verhältnis von fläche zur gesamtfläche
- ziehe sample (mit uniformen sample) aus dieser Verteilung um lightsource aus der gruppe auszuwählen

Light Source Area Sampling

- ziehe 2 uniforme sample um auf der Fläche der Lightsource uniform ein sample zu ziehen
- überprüfe ob andere Lightsource aus gruppe zwischen p und gesampelten punkt sp liegt und setze sp auf schnittpunkt
- berechne richtungsvektor zu p
- berechne pdf für ganze gruppe
 - gewichte pdf der einzelnen Lightsources aus gruppe mit ihrem anteil an der gesamt Fläche
- teste ob p von sp aus sichtbar ist, wenn nein setze farbe für punkt auf 0

Light Source Area Sampling

- gewichte lichtintensität von licht auf sp mit pdf (wenn gruppe dann gruppen pdf)
- sammle lichtfarbe von gesampletem objekt
- berechne licht menge mit intensität * (gesamt)fläche * π
- summiere lichtmenge auf

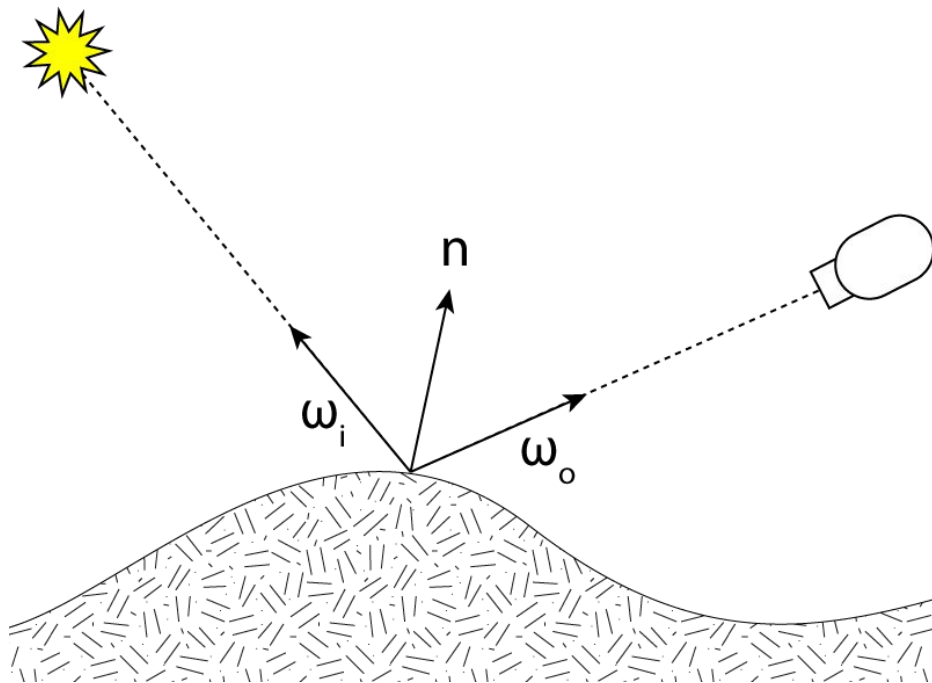
nach sample vorgang:

- berechne gesamte licht farbe durch lichtfarbe für alle lichter * intensität für alle Lichter gewichtet mit gesamt Lichtmenge

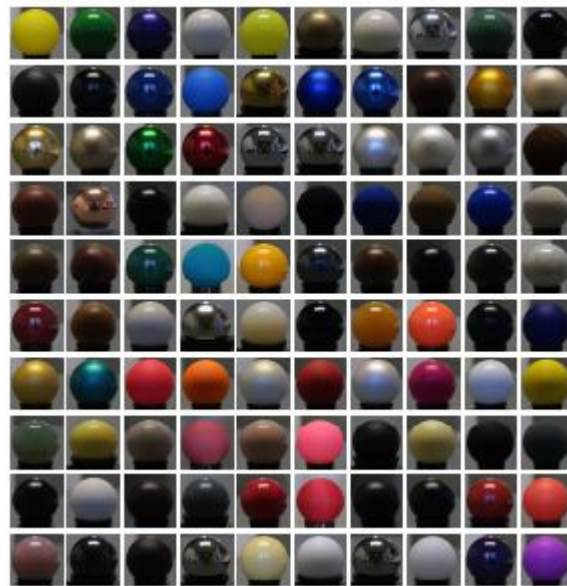
Light Source Area Sampling

- normalisiere gesamt lichtmenge durch sampleanzahl
- berechne farbe für p durch: farbe des objektes* gesamt lichtfarbe* lichtmengen summe
- Hier tritt problem auf das helle lichter clippen (überstrahlung) und dunkle lichter zu dunkel sind
- durch normalisieren mit dem logarithmus durch einen faktor werden helle lichtquellen auf den faktor beschränkt und dunkle lichtquellen aufgehellt
- pdf konvertiert pdf über lightsource fläche zu pdf von raumwinkel von p zur lightsource mit:
 - abstand^2 zwischen sp und p / $\text{winkel zwischen normale und sp} * \text{fläche der lightsource}$

BRDF Sampling

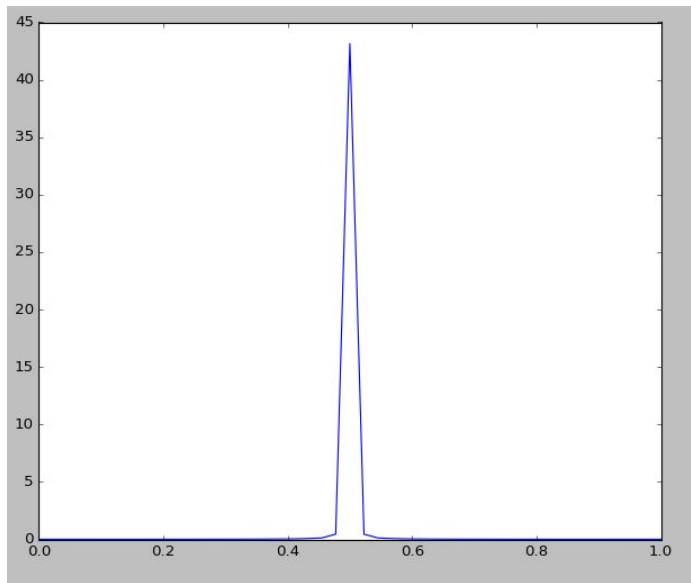


BRDF Funktion mit 4 Winkeln

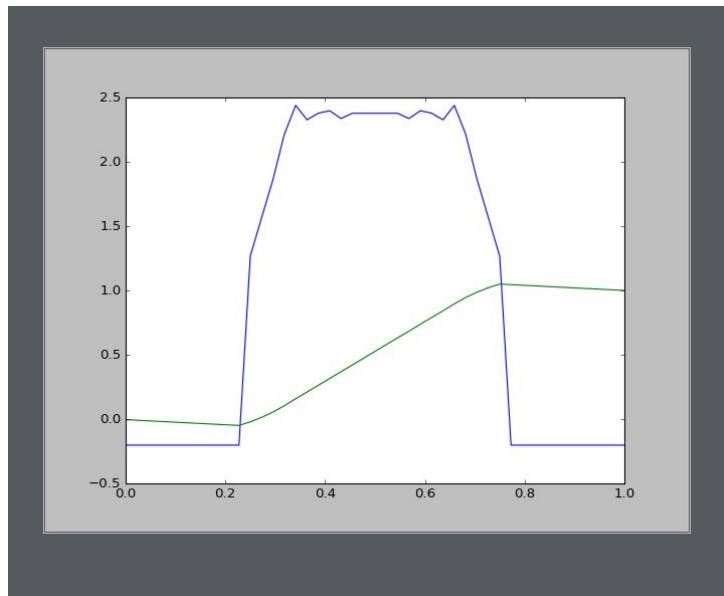


MERL Database

BRDF Sampling



Sehr glänzende, harte Oberfläche



Weiche, diffuse Oberfläche

BRDF Sampling

- Vorbereitung der PDFs für Theta, Phi
- Überlagerung der PDFs für alle Lichtquellen (Gewichtet nach Fläche der Lichtquelle)
- Samples ziehen aus PDFs (orientiert gegen die Schnittnormale)
- In die Szene schießen
- Gesammeltes Licht gewichten nach Phi und Theta PDF

Licht addieren, interagieren lassen

Ansatz: Lichtquelle mit höherer Intensität hat mehr Einfluss auf Lichtfarbe

Für jedes Sample wird gesammelte Lichtfarbe, Intensität gespeichert

Nach Sampling Intensitäten summieren, Anteil der einzelnen Intensität zum Ganzen ist Gewicht

Multiplikation von Licht- und Objektfarbe ergibt physikalisch “korrektes” Verhalten

Hat Multiple Importance Sampling etwas gebracht?

Ja.

Was verbessert werden kann

- Implementation in C++
- Schnellere Intersections mit Octree / Optimierten Ray Intersection Algorithmus
- Korrekte behandlung der BRDF Werte
- Rekursives Rendering nicht nur Direct Light

