

# Berechnung eines optisch realistischen Schattens transparenter Okkluder in Echtzeit

Mario König

TH Köln

Bachelorarbeit vom 09.09.2016

Studiengang Medientechnologie

Betreuer: Prof. Dr. Arnulph Fuhrmann

**Abstract:** Diese Bachelorarbeit stellt Theorie und Implementation zweier Verfahren vor, die die realistische Darstellung von Schatten transparenter Verdecker in Echtzeit ermöglichen. Diese werden in Hinsicht auf Performance und optischen Eindruck evaluiert.

**Keywords:** Echtzeit, Rendering, Transparenz, Schatten

## 1 Einleitung

Schatten sind ein wichtiger Bestandteil bei der Erzeugung realistischer digitaler Bilder. Dennoch werden bei ihrer Berechnung in Echtzeit häufig Kompromisse eingegangen, um lange Renderzeiten zu vermeiden. Einer dieser Kompromisse ist der Verzicht auf korrekte Schatten transparenter Verdecker, die normalerweise entweder als komplett opak, oder gar nicht dargestellt werden. Diese Arbeit implementiert und evaluiert zwei aktuelle Verfahren, die in der Lage sind für semi-transparente Okkluder sowohl wellenlängenabhängige Transmission, als auch Lochmaskentransparenz (partial coverage) optisch korrekt darzustellen. Dies sind zum einen eine Erweiterung von Moment Shadow Mapping [PMWK16] und zum anderen ein neuer Algorithmus, Linked-List Shadows, der ohne Erstellung einer Shadowmap auskommt.

## 2 Umsetzung

Moment Shadow Mapping benötigt zunächst sortierte Primitive für korrekte, farbige Transparenz, was mit Hilfe GPU-basierter verketteter Listen [YHGT10] erreicht wird. Es ergibt sich pro Farbkanal jeweils eine Shadowmap, in der die Momente gespeichert werden. Die Auswertung erfolgt mit Hilfe desselben Algorithmus, den Moment Shadow Mapping verwendet [PK15]. Dieser berechnet eine möglichst genaue, untere Grenze der Schattenintensität.

Für Linked-List Shadows werden verkettete Listen zur Speicherung aller in der Szene vorhandenen semi-transparenten Primitive verwendet. Es wird jedoch keine echte Shadowmap erzeugt. Stattdessen werden die Listen pro Fragment mit Hilfe von Insertion-Sort sortiert und direkt zur Schattenberechnung traversiert. Dabei erfolgt die Berechnung der Intensität über Blending der Fragmente mit dem under-Operator [PD84].

### 3 Evaluation

In der Auswertung ergibt sich, dass beide Verfahren auf einem Testsystem mit einer Nvidia GTX 980 im echtzeitfähigen Bereich bleiben. Beispielsweise beträgt die Renderzeit bei 1024<sup>2</sup> Pixeln Shadowmap Auflösung und 1080p in einer Szene mit über 300.000 semi-transparenten Dreiecken ca. 8ms für transparentes Moment Shadow Mapping und ca. 12ms mit dem Linked List Verfahren. Auch qualitativ zeigen beide Verfahren gute Ergebnisse; Light Leaking kann im Vergleich zu Exponential Variance Shadow Maps mit Moment Shadow Maps reduziert, mit dem Linked List Verfahren beseitigt werden. Nachteilig ist bei beiden Verfahren die erhöhte Renderzeit, sowie vor allem der sehr hohe Speicherverbrauch.

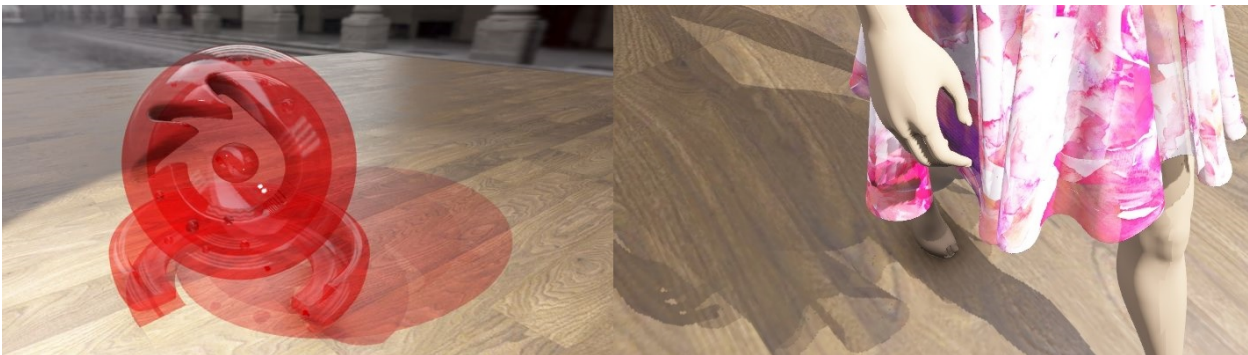


Abbildung 1: Links: Linked-List Shadows Verfahren mit Transmission und Lochmaskentransparenz. (Blender Material Preview Geometrie von Robin “tuqueque” Marín (CC-BY)). Rechts: Transparentes Moment Shadow Mapping mit ausschließlich Lochmaskentransparenz.

### Literatur

- [PD84] Thomas Porter and Tom Duff. Compositing digital images. In *ACM Siggraph Computer Graphics*, volume 18, pages 253–259. ACM, 1984.
- [PK15] Christoph Peters and Reinhard Klein. Moment shadow mapping. In *Proceedings of the 19th Symposium on Interactive 3D Graphics and Games*, pages 7–14. ACM, 2015.
- [PMWK16] Christoph Peters, Cedrick Munstermann, Nico Wetzstein, and Reinhard Klein. Beyond hard shadows: moment shadow maps for single scattering, soft shadows and translucent occluders. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 159–170. ACM, 2016.
- [YHGT10] Jason C Yang, Justin Hensley, Holger Grün, and Nicolas Thibieroz. Real-time concurrent linked list construction on the gpu. In *Computer Graphics Forum*, volume 29, pages 1297–1304. Wiley Online Library, 2010.