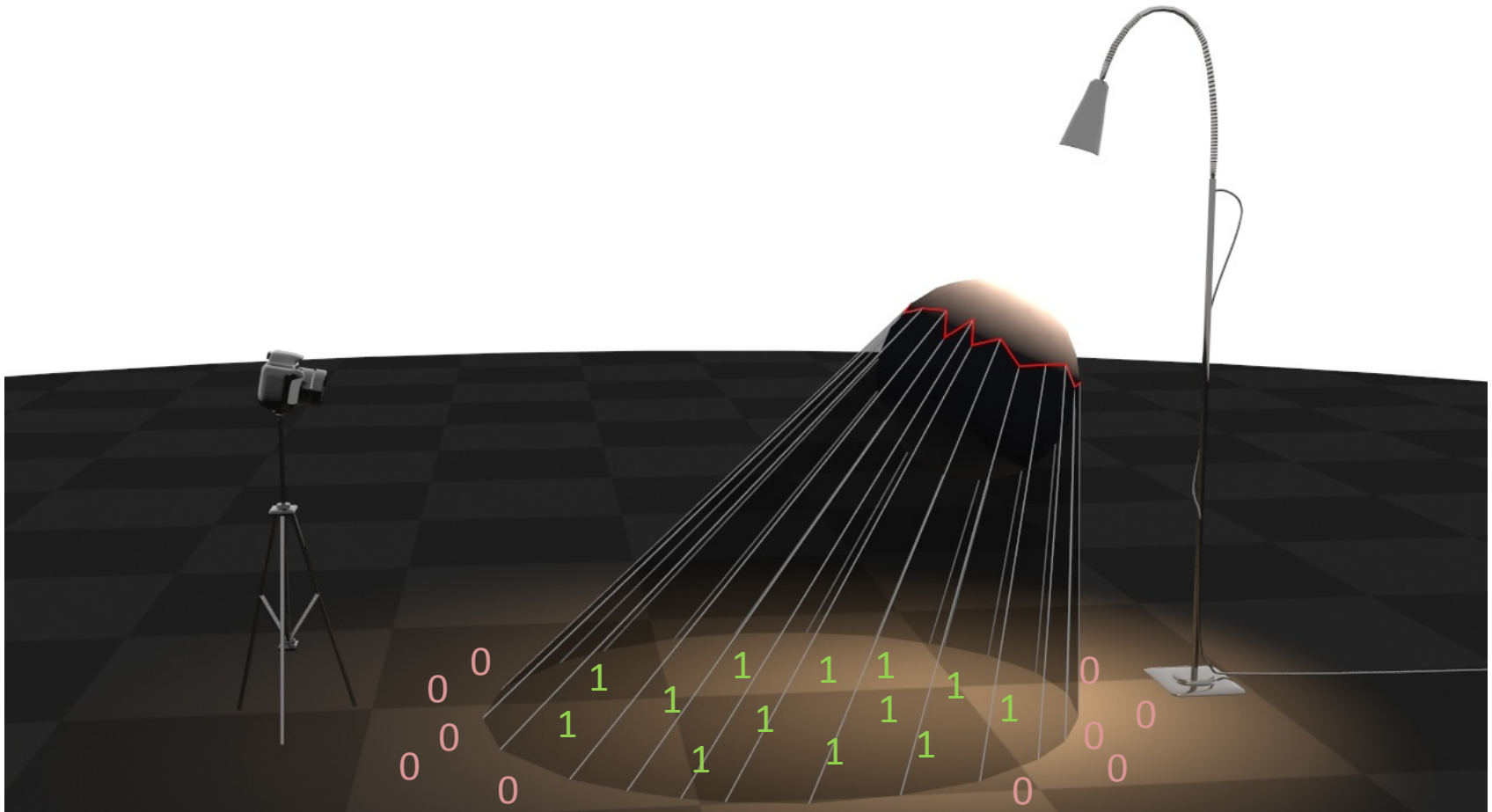


Übung Interaktive Computergrafik

14.05.2014

- ▶ Schattenvolumen mit Geometry-Shader
- ▶ Ein Schattenvolumen pro Dreieck statt nur Silhouette
- ▶ Artefakte an Rändern der Dreiecks-Schattenvolumen

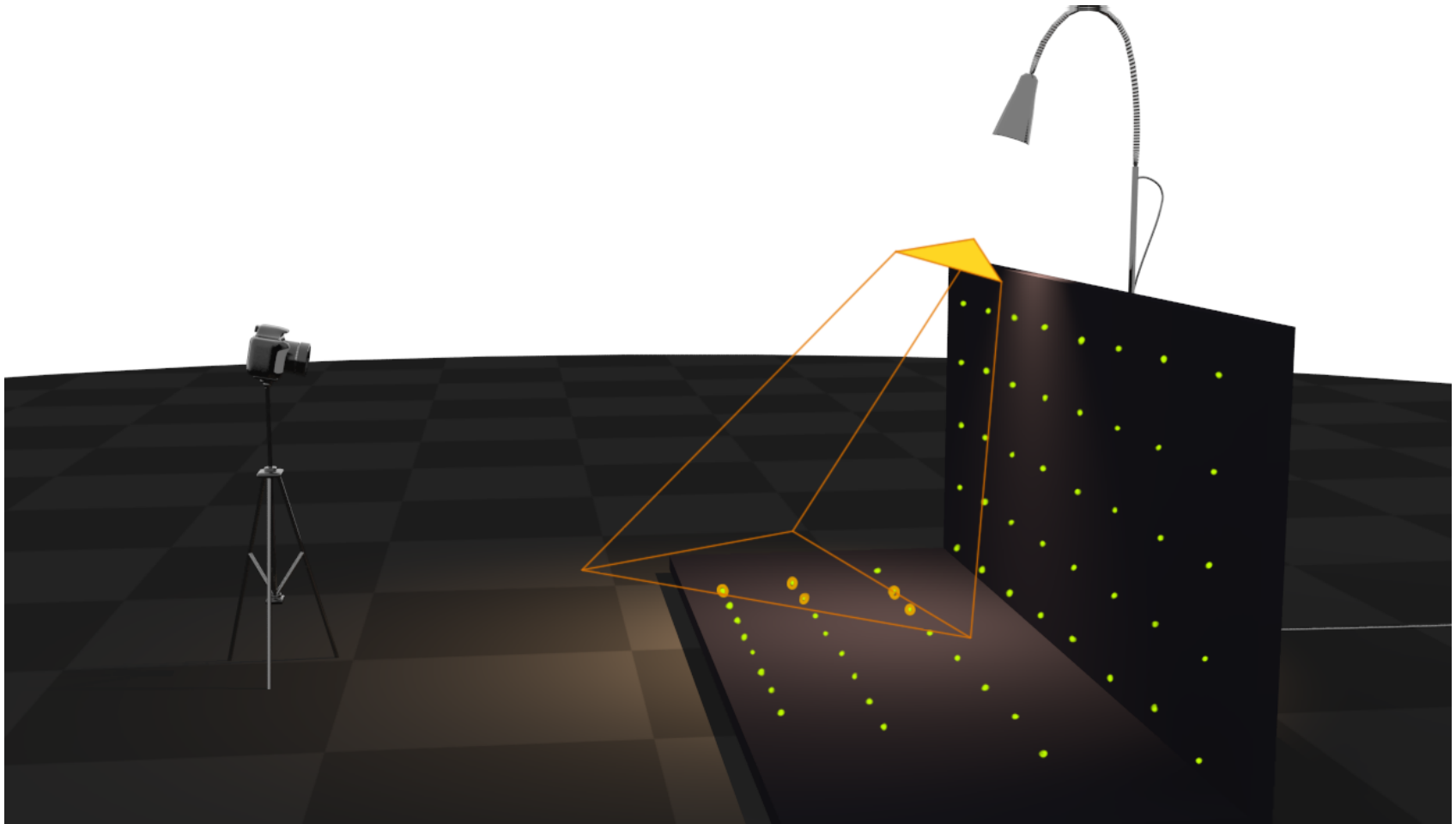
Ausblick: Aktuelle Forschung und Shadow Volumes





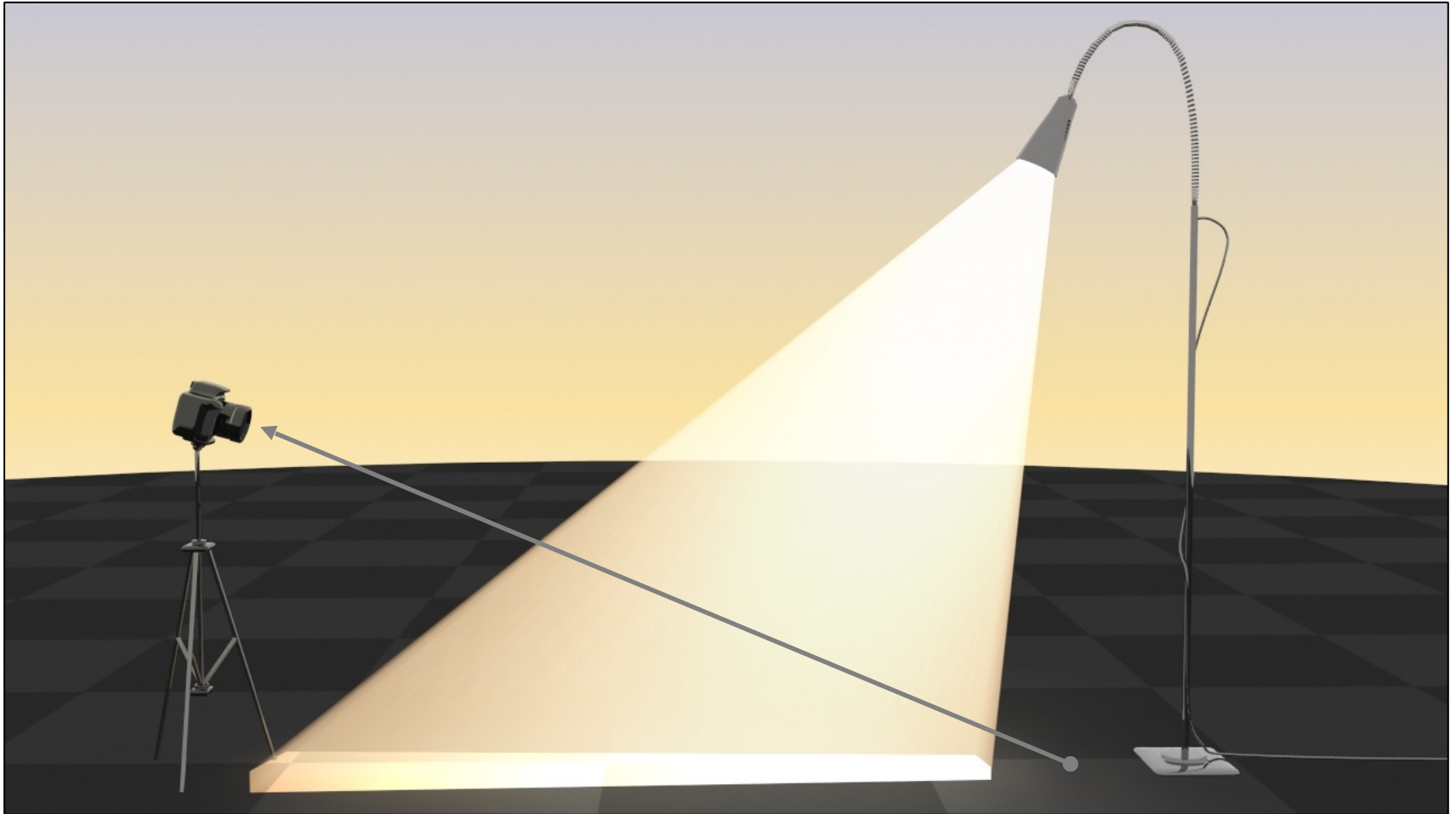
Ausblick: Aktuelle Forschung und Shadow Volumes

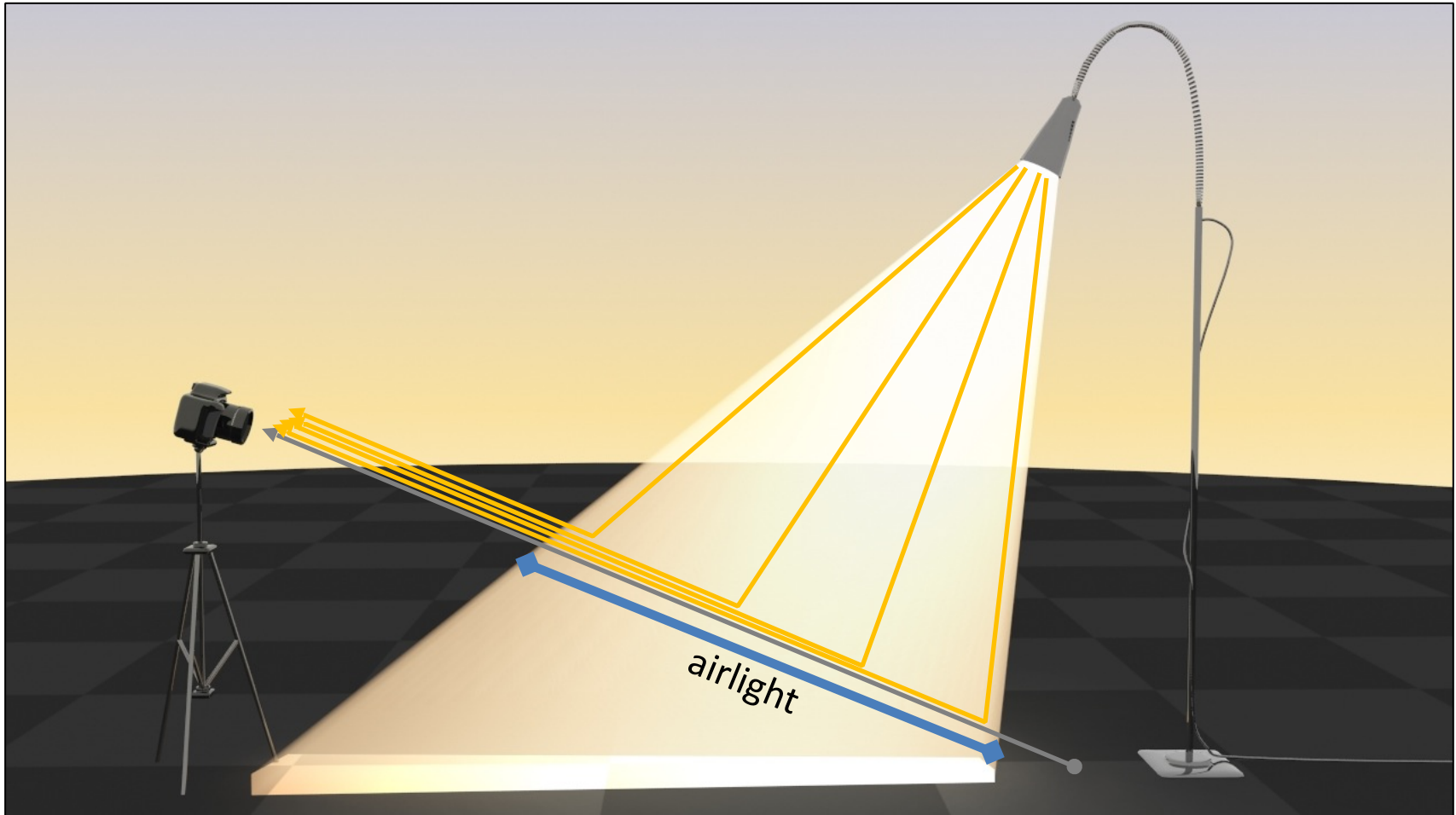
An Efficient Alias-free Shadow Algorithm for Opaque and Transparent Objects using per-triangle Shadow Volumes, Erik Sintorn, Ola Olsson, Ulf Assarsson, *ACM SIGGRAPH Asia 2011*



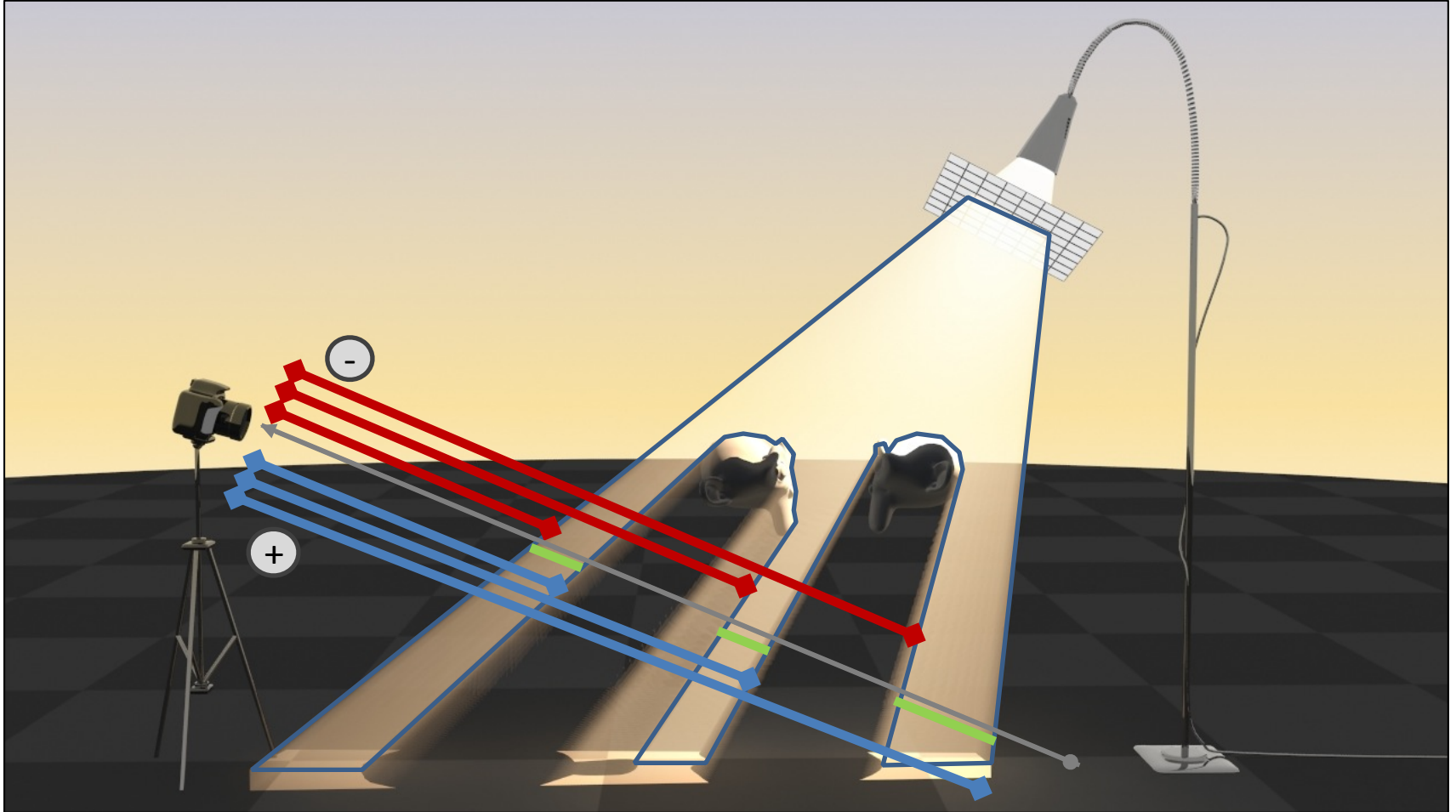
Ausblick: Aktuelle Forschung und Shadow Volumes

Volumetric Shadows using Polygonal Light Volumes, Markus Billeter, Erik Sintorn, Ulf Assarson, *High Performance Graphics 2010*





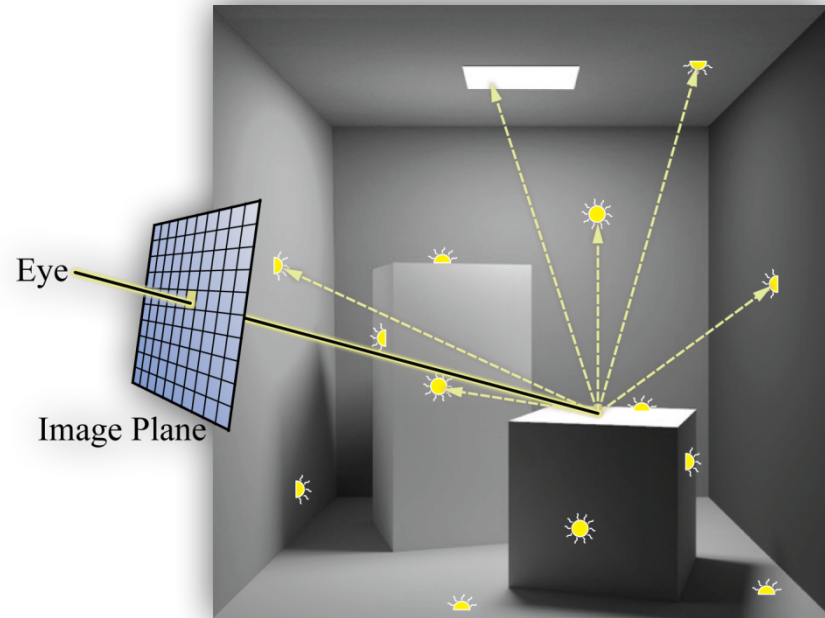
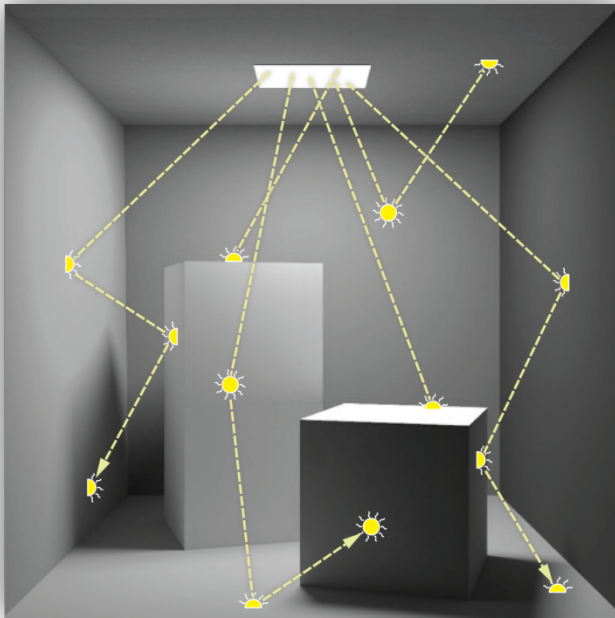
Ausblick: Aktuelle Forschung und Shadow Volumes



Ausblick: was tun, wenn man viele LQ hat?

Instant Radiosity

- ▶ Instant Radiosity und sog. Many-Light Methoden approximieren indirekte Beleuchtung durch eine Menge von Punktlichtquellen
- ▶ für jede dieser LQ benötigen wir Schattentests, z.B. je eine Shadow Map



Ausblick: Imperfect Shadow Maps



- ▶ die Idee geht zurück auf eine Studie („Perceptual Influence of Approximate Visibility in Indirect Illumination“), die untersucht hat
 - ▶ wie akkurat muss Sichtbarkeit für indirekte Beleuchtung überhaupt berechnet werden?
 - ▶ welche Approximationen funktionieren wie gut?
 - ▶ wie schlimm ist es, wenn bei einer Shadow Map X% der Tiefenwerte auf zufällige Werte gesetzt werden?



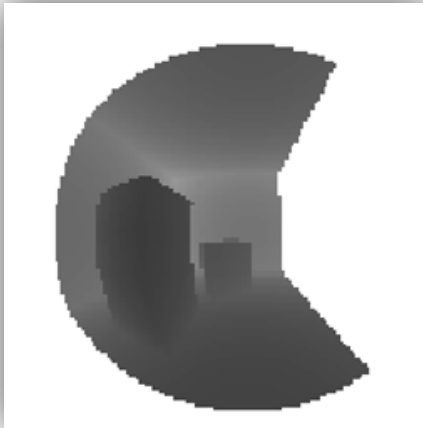
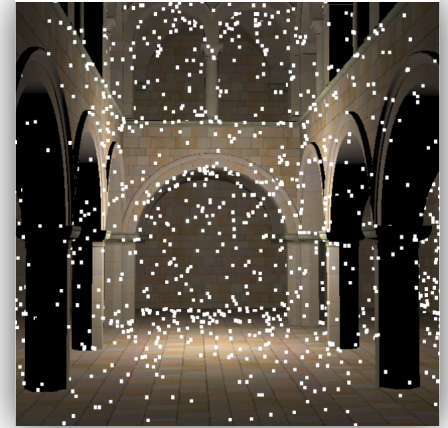
akkurate Tiefenwerte



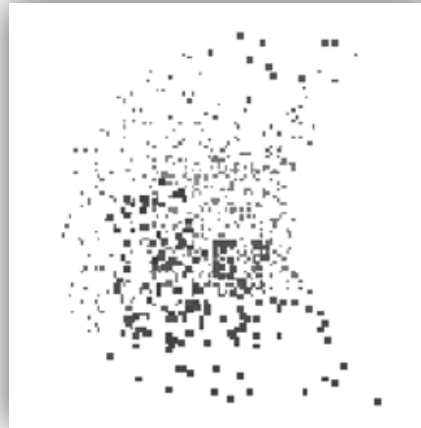
20% der Tiefenwerte fehlerhaft

Ausblick: Imperfect Shadow Maps

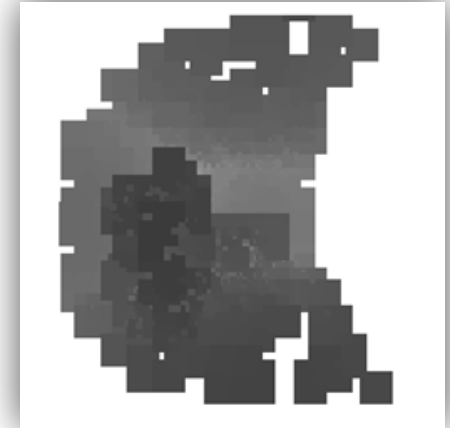
- ▶ solche fehlerhaften Shadow Maps kann man sehr viel einfacher erzeugen als korrekte
- ▶ Idee: repräsentiere (für das SM Rendering) die Szene nicht durch Dreiecke, sondern durch einige tausend Punkte auf den Oberflächen
- ▶ das Resultat (Mitte) ist unbrauchbar, aber: mit einer heuristischen Bildraum-Operation lassen sich die Lücken größtenteils schließen (die SM ist aber immer noch fehlerhaft)



normale Shadow Map
(gezeichnet mit Dreiecken)

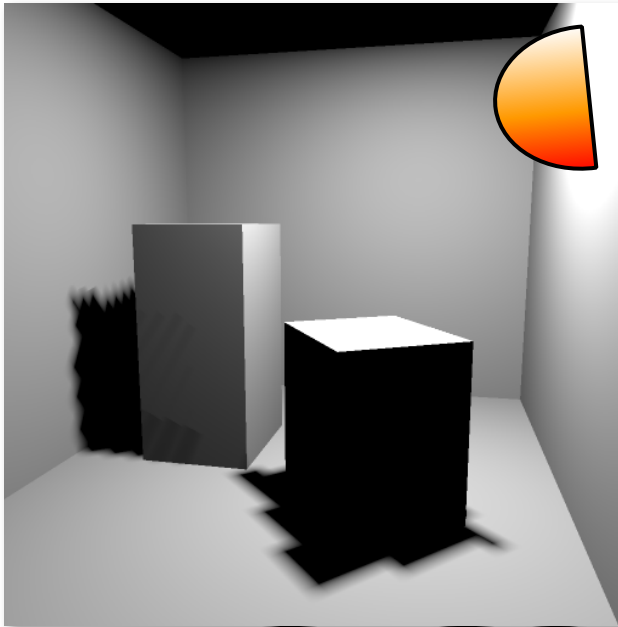


Punktrendering
(nähere Punkte größer)

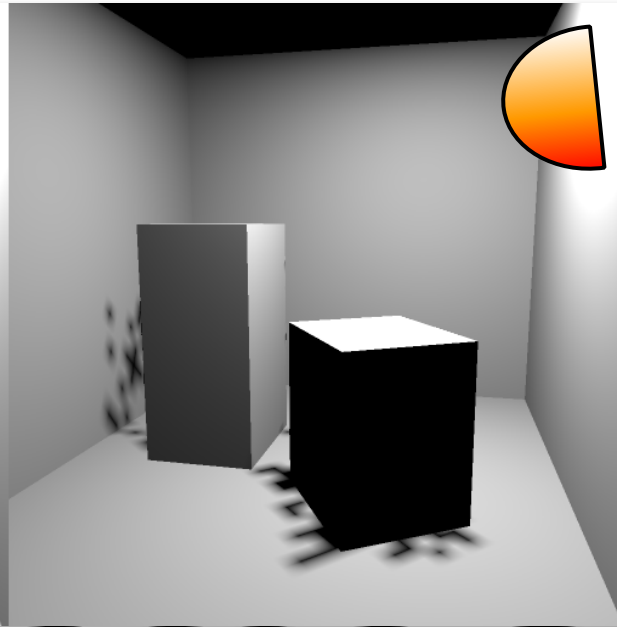
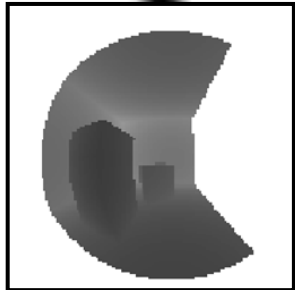


imperfekte Shadow Map
„pull-push hole filling“

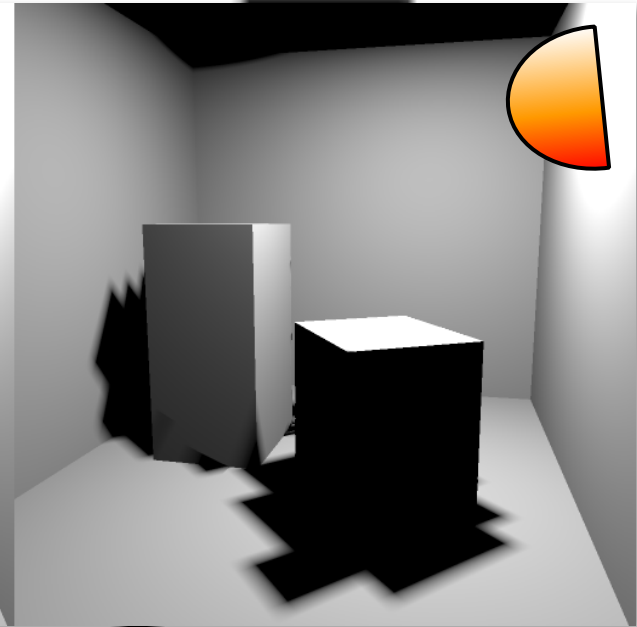
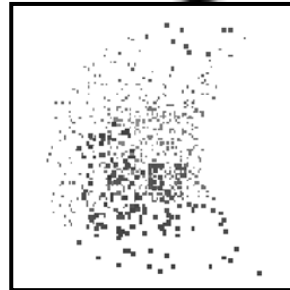
Ausblick: Imperfect Shadow Maps



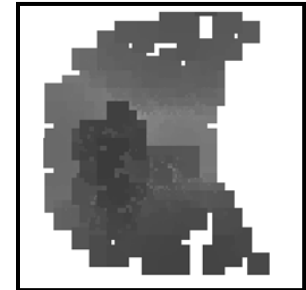
Referenz



ohne pull-push

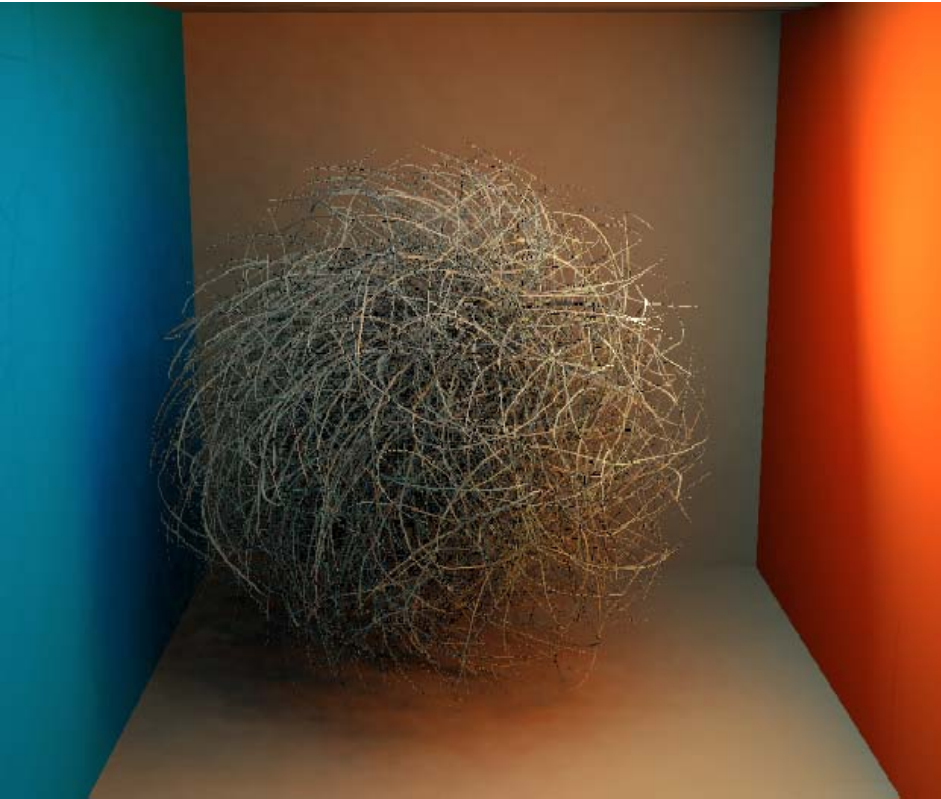


mit pull-push

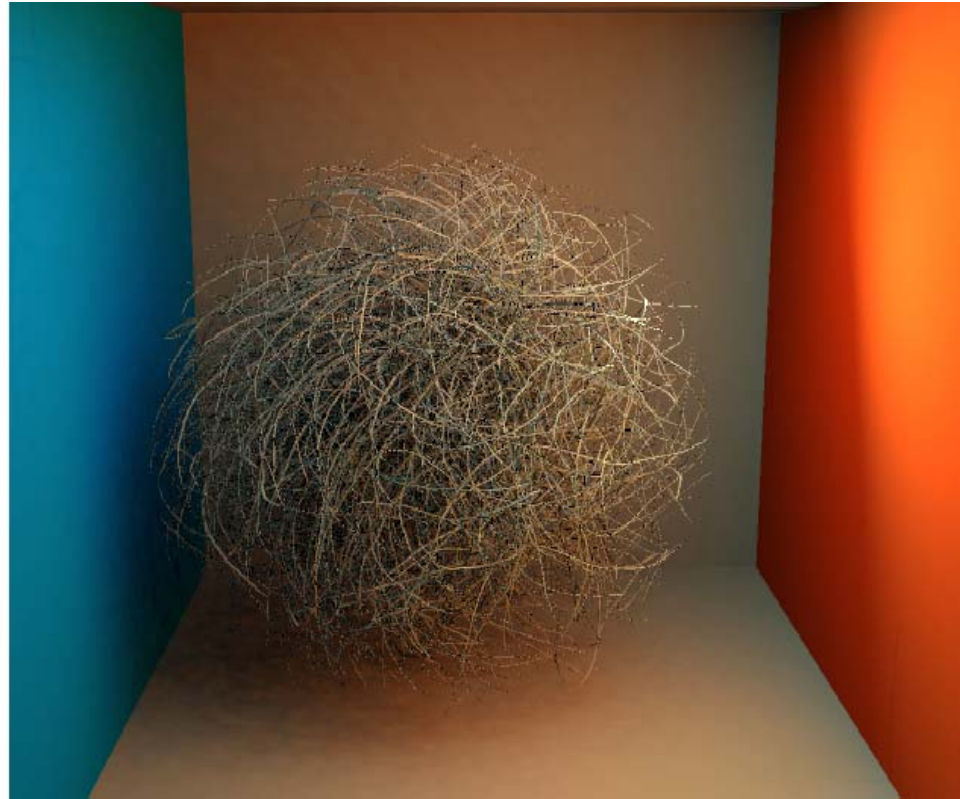


Ausblick: Imperfect Shadow Maps

- ▶ diese Idee funktioniert, in erster Linie deswegen, weil sich die Fehler durch die große Anzahl Lichtquellen „herausmitteln“



Referenz
(normale Shadow Maps)

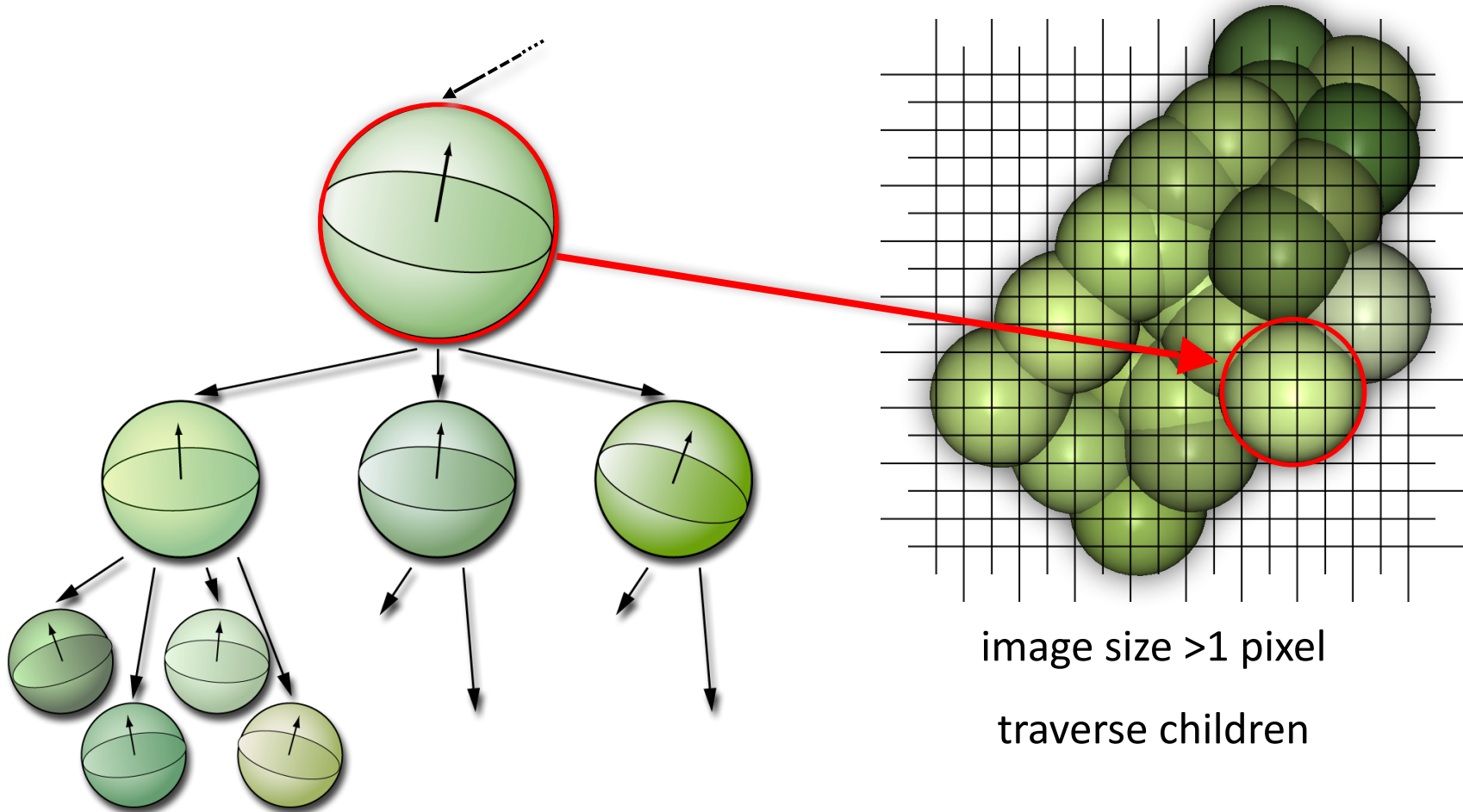


Imperfect Shadow Maps

Ausblick: was tun, wenn man viele LQ hat?

High-Quality Point-based Rendering

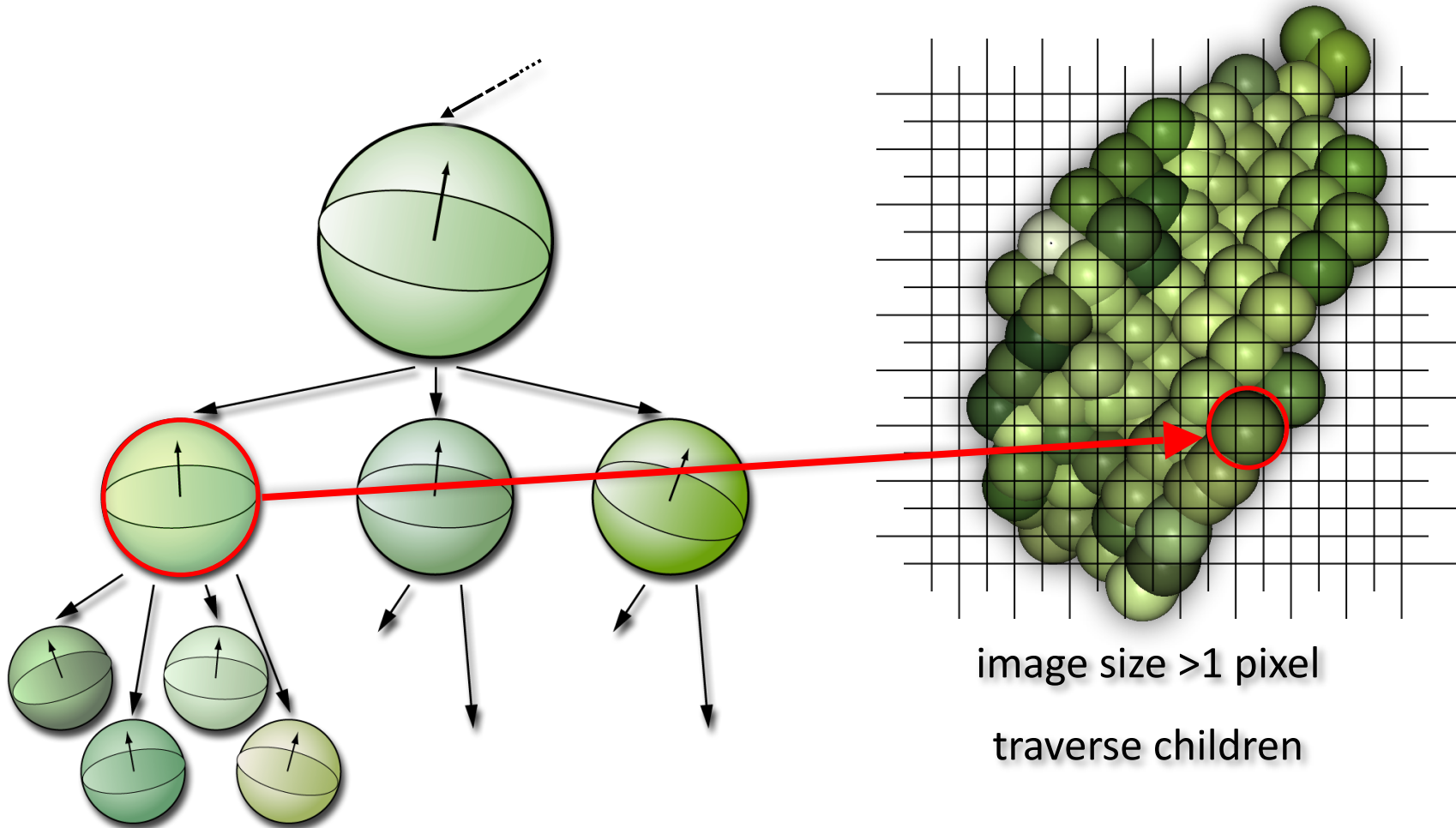
► man kann mit Punktrepräsentationen (wenn es sein muss) auch qualitativ bessere Darstellungen erreichen



Ausblick: was tun, wenn man viele LQ hat?

High-Quality Point-based Rendering

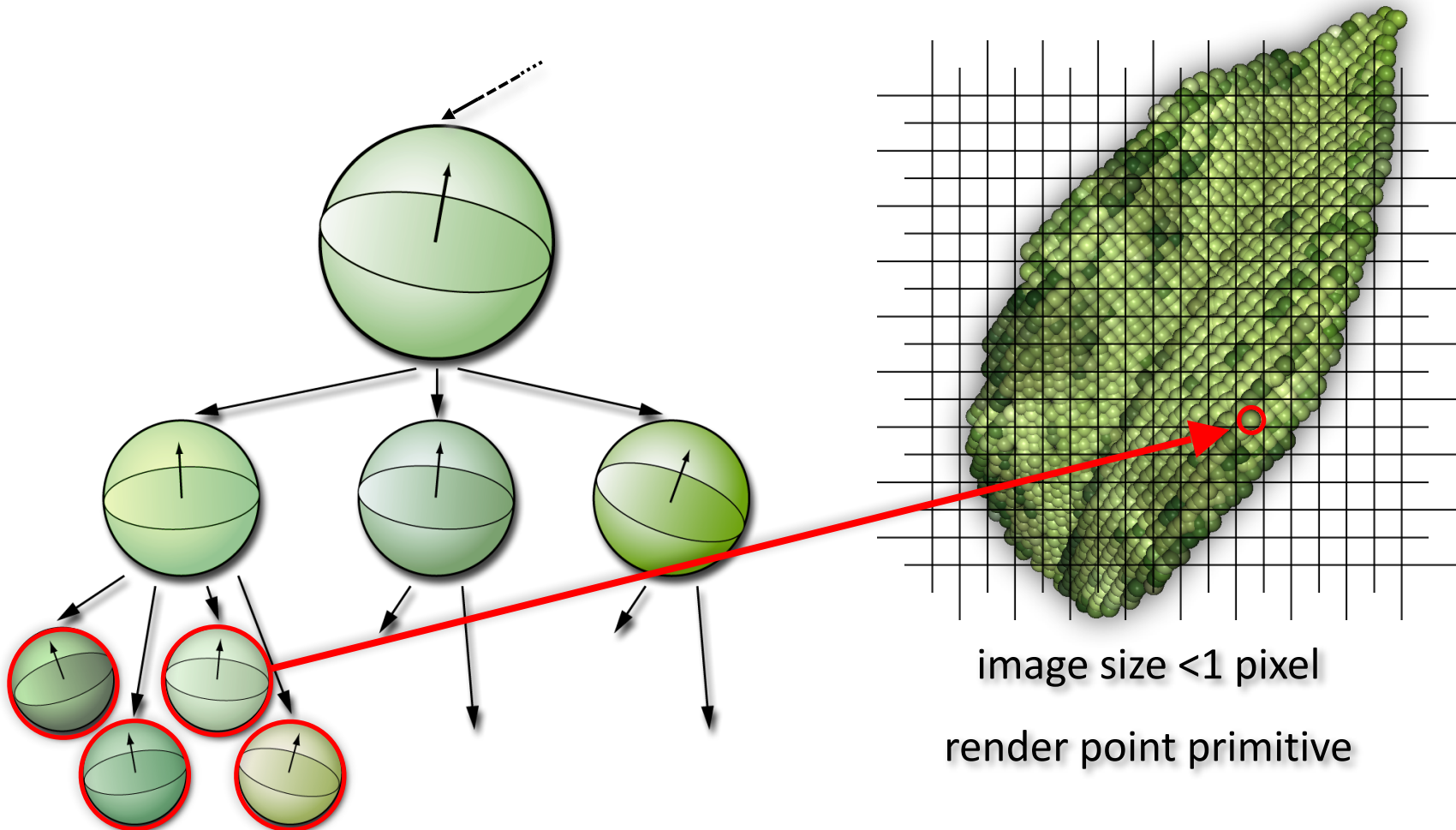
- ▶ man kann mit Punktrepräsentationen (wenn es sein muss) auch qualitativ bessere Darstellungen erreichen



Ausblick: was tun, wenn man viele LQ hat?

High-Quality Point-based Rendering

▶ man kann mit Punktrepräsentationen (wenn es sein muss) auch qualitativ bessere Darstellungen erreichen



Ausblick: was tun, wenn man viele LQ hat?

Micro-Rendering

- ▶ akkurate Environment Maps / Tiefenkarten mittels Punkthierarchien
- ▶ eigentlich für Final Gathering entwickelt, implementiert CUDA/OpenCL
- ▶ Performanz in 2009: $\sim 16k$ in 100 ms, je 24^2 Pixels

