CS5630 Physically Based Realistic Rendering

Steve Marschner

Spring 2026

01 Introduction

Physically Based Realistic Rendering

Physically Based

- The goal is for the image to be accurate
- The approach is to simulate how the image was formed

Realistic

- The goal is for the image to look real

Rendering

- The input is a scene
- The output is a depiction of the scene

CS 5630

Covers the theory and practice of physics based rendering

- Monte Carlo integration
- Physics of light transport
- Path tracing algorithms
- Various advanced methods, in less detail

Centers around assignments and projects

- Notebook assignments to explore math
- Programming assignments to implement a renderer
- Final project leading to a competition for best rendered image

Introductions

Steve Marschner

- Instructor, Professor of CS specializing in graphics
- Research focus on accurate material models (see 2004 Technical Oscars…)
- Full disclosure: I also work for NVIDIA Research

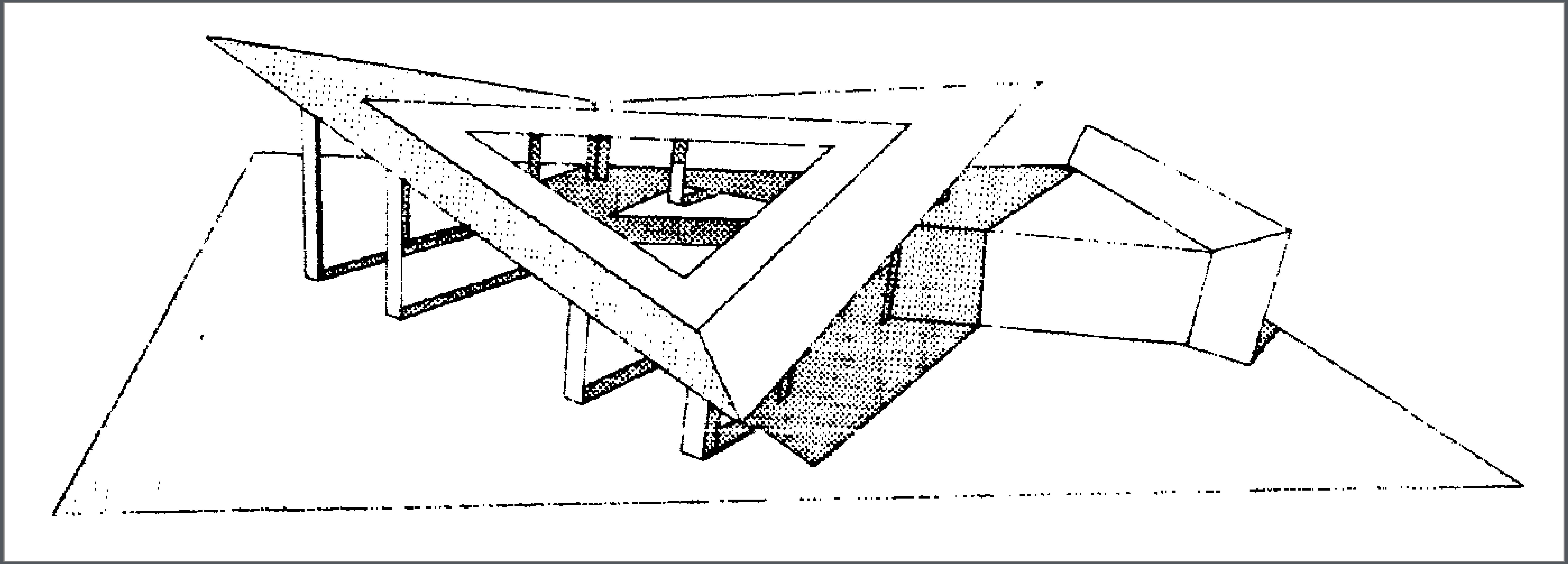
Mariia (Masha) Soroka

- PhD TA, research area = rendering and differentiable rendering

Gemmechu Hassena

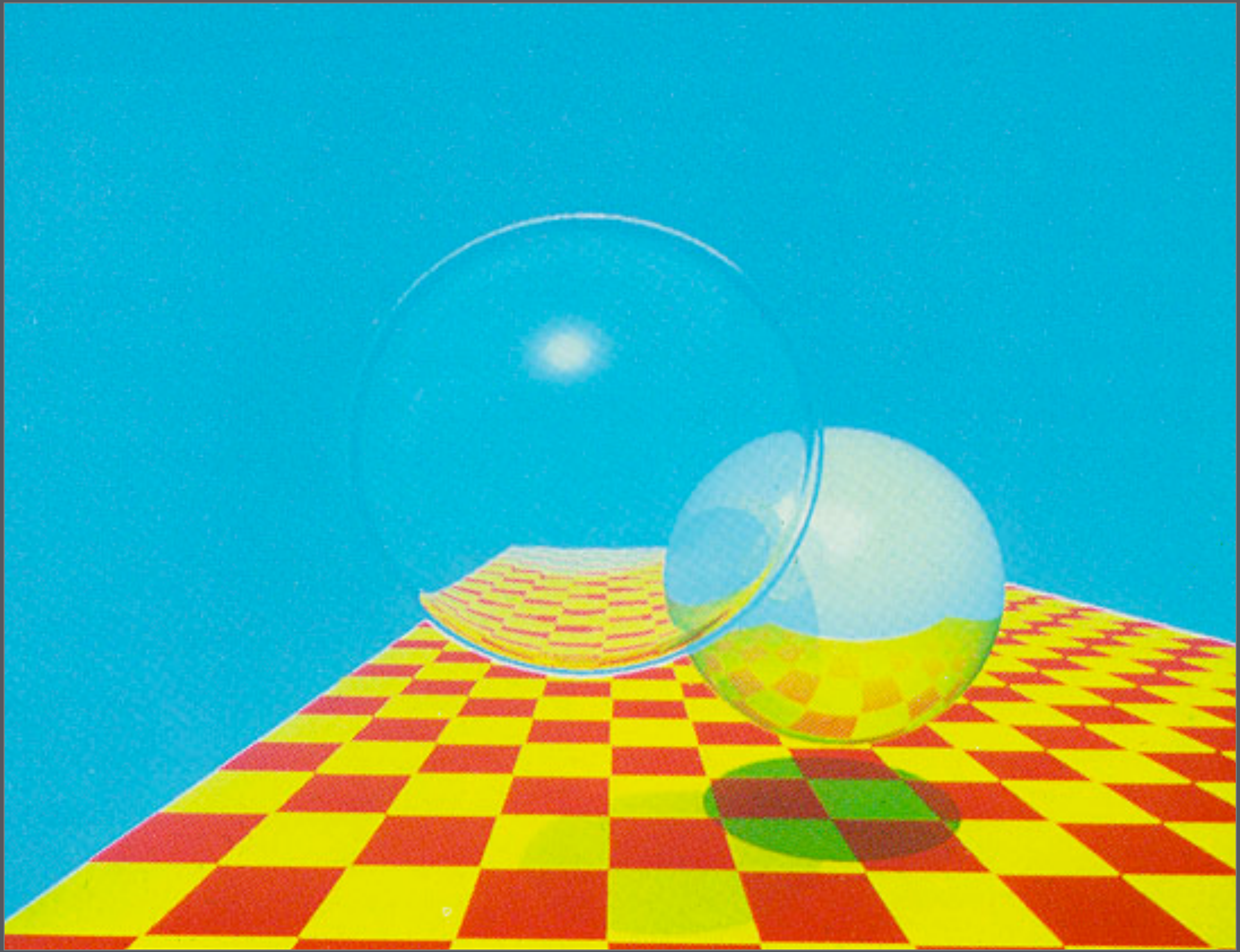
- PhD TA, research area = vision-based realistic scene reconstruction

Ray Tracing

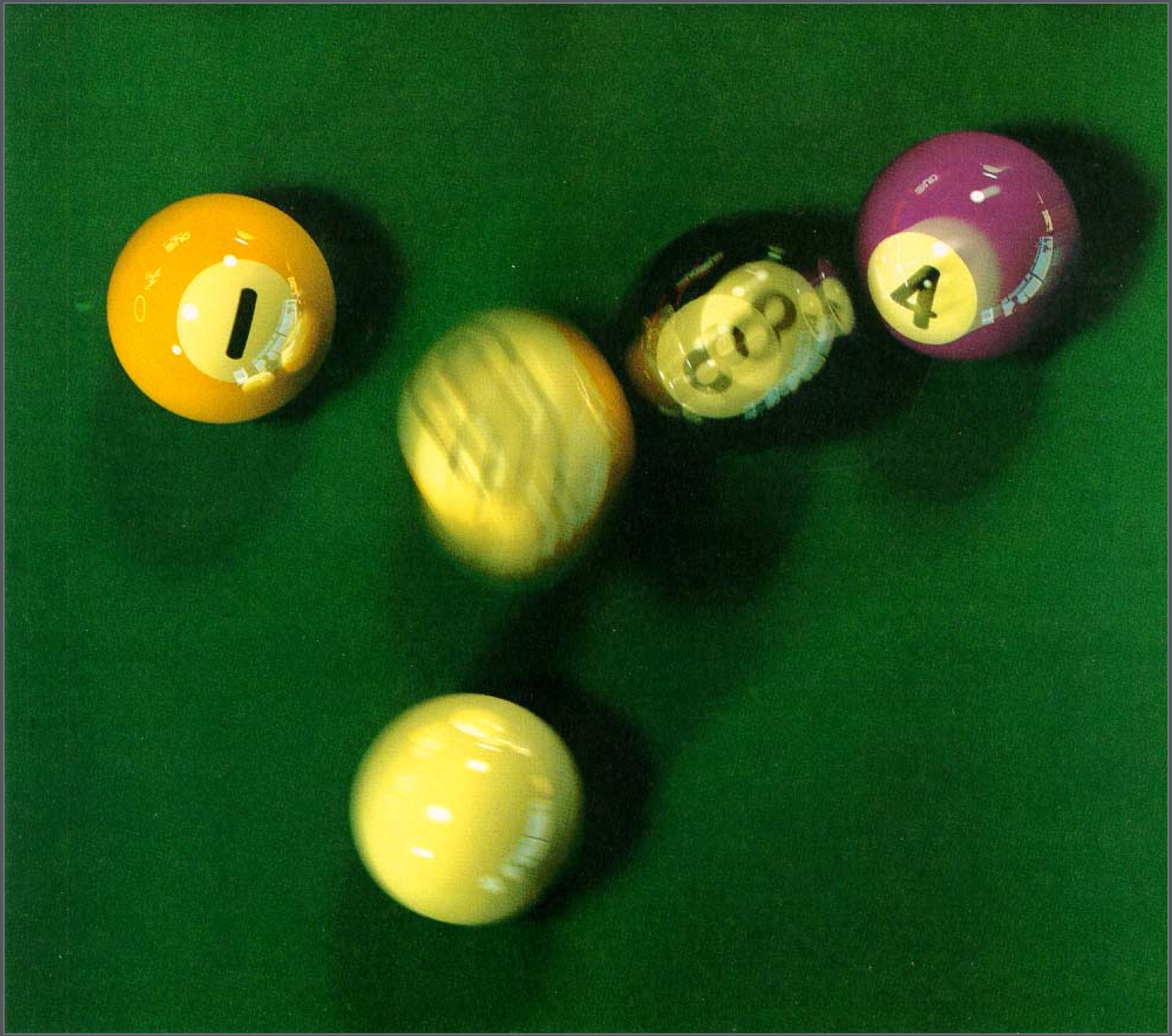


Appel 1968

Ray Tracing for shadows



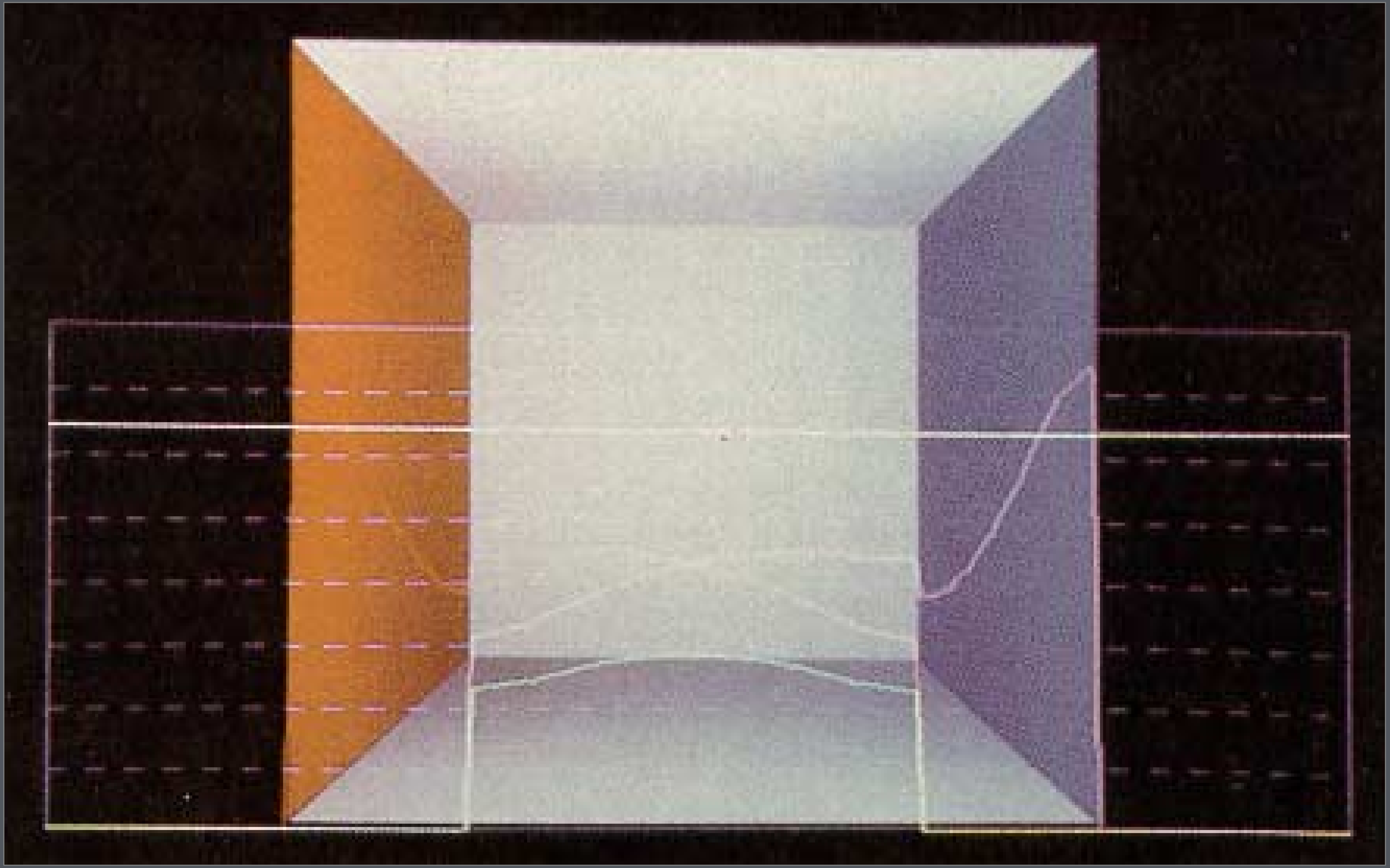
Whitted 1980
Recursive ray tracing



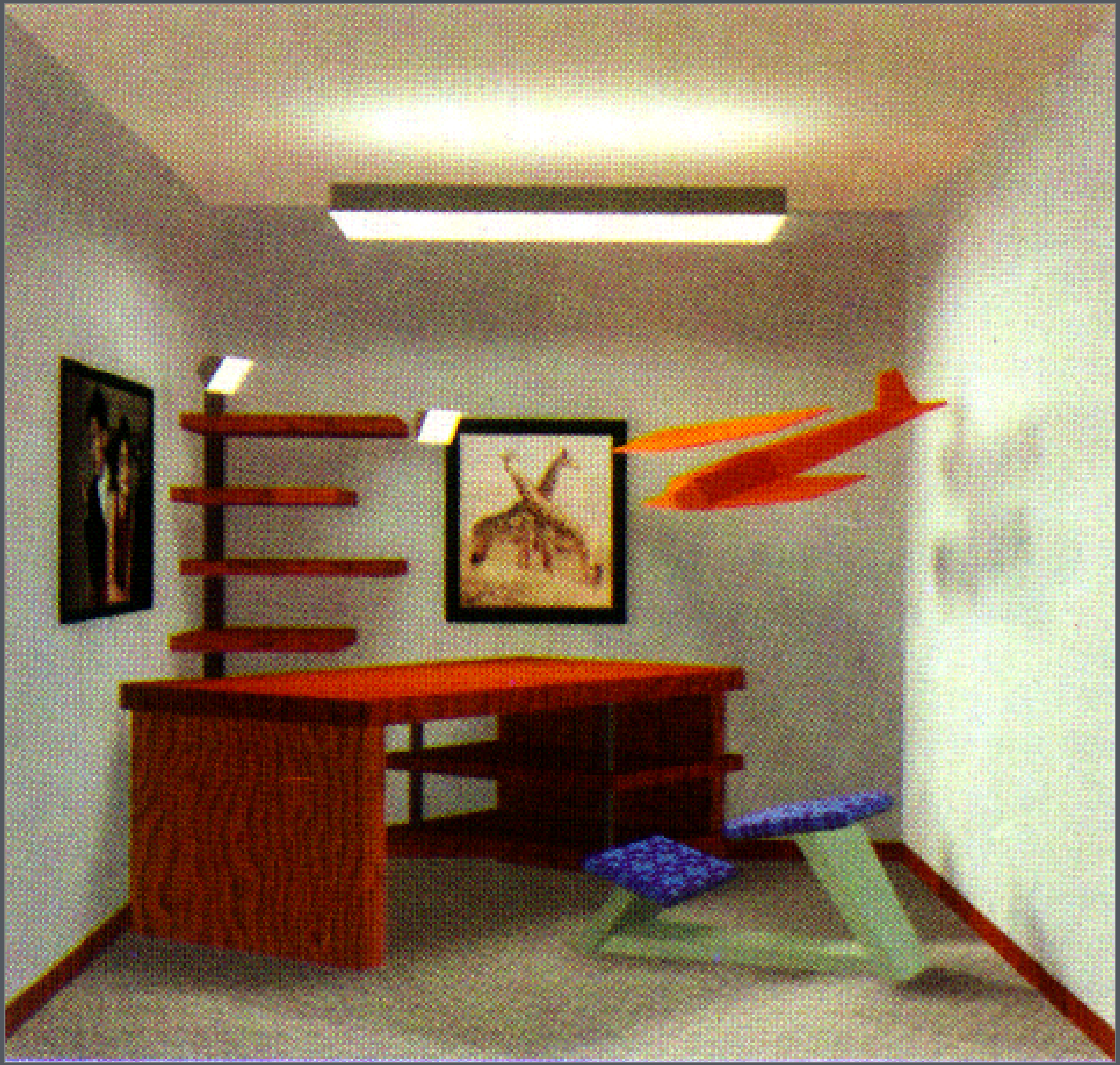
Cook, Porter, Carpenter 1984

Distribution Ray Tracing

Radiosity



Goral et al. 1984
Radiosity method



Hanrahan et al. 1991
Hierarchical radiosity

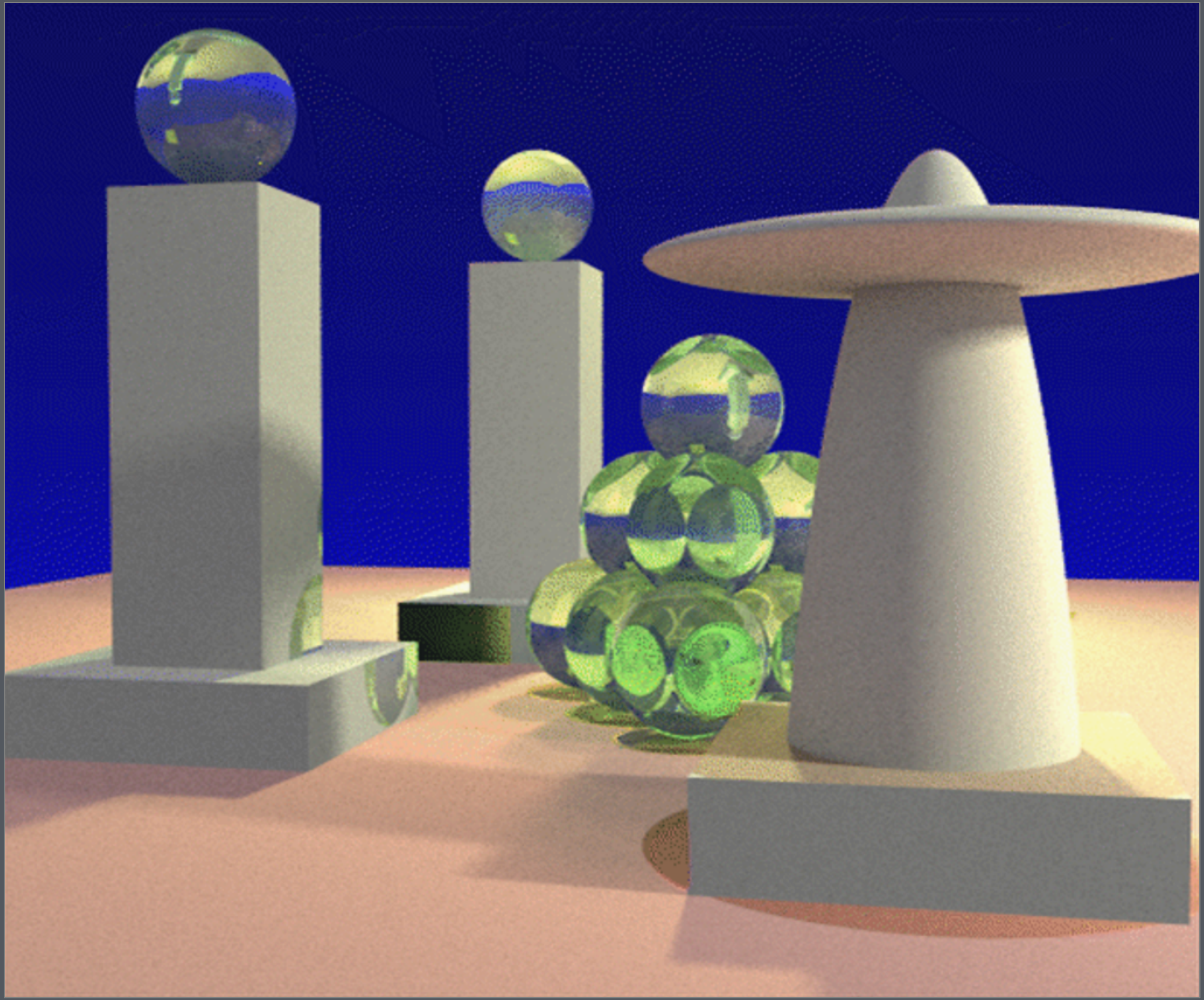
Lischinski et al. 1993

Discontinuity meshing

Sillion et al. 1991

Nondiffuse radiosity

Path Tracing

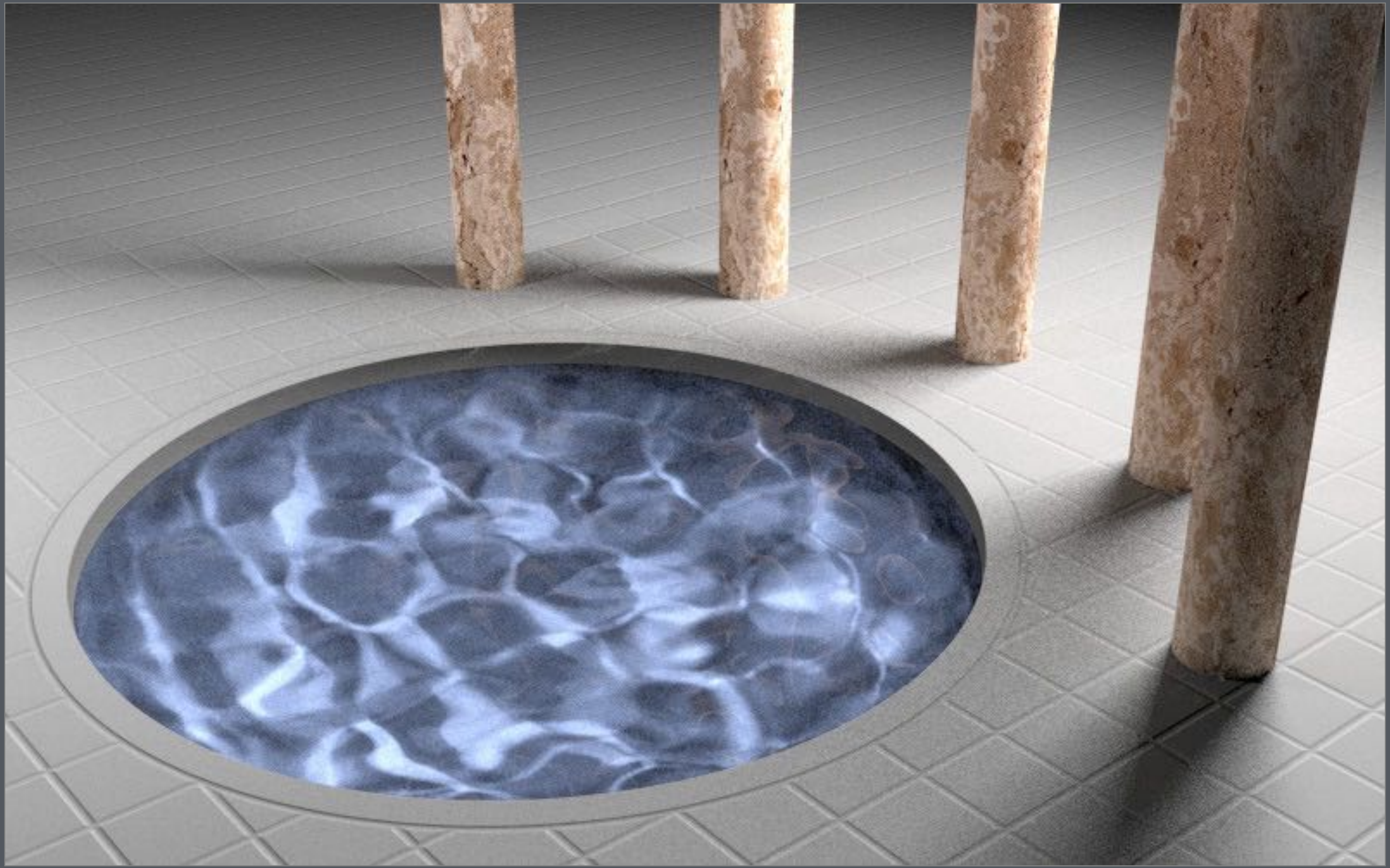


Kajiya 1986

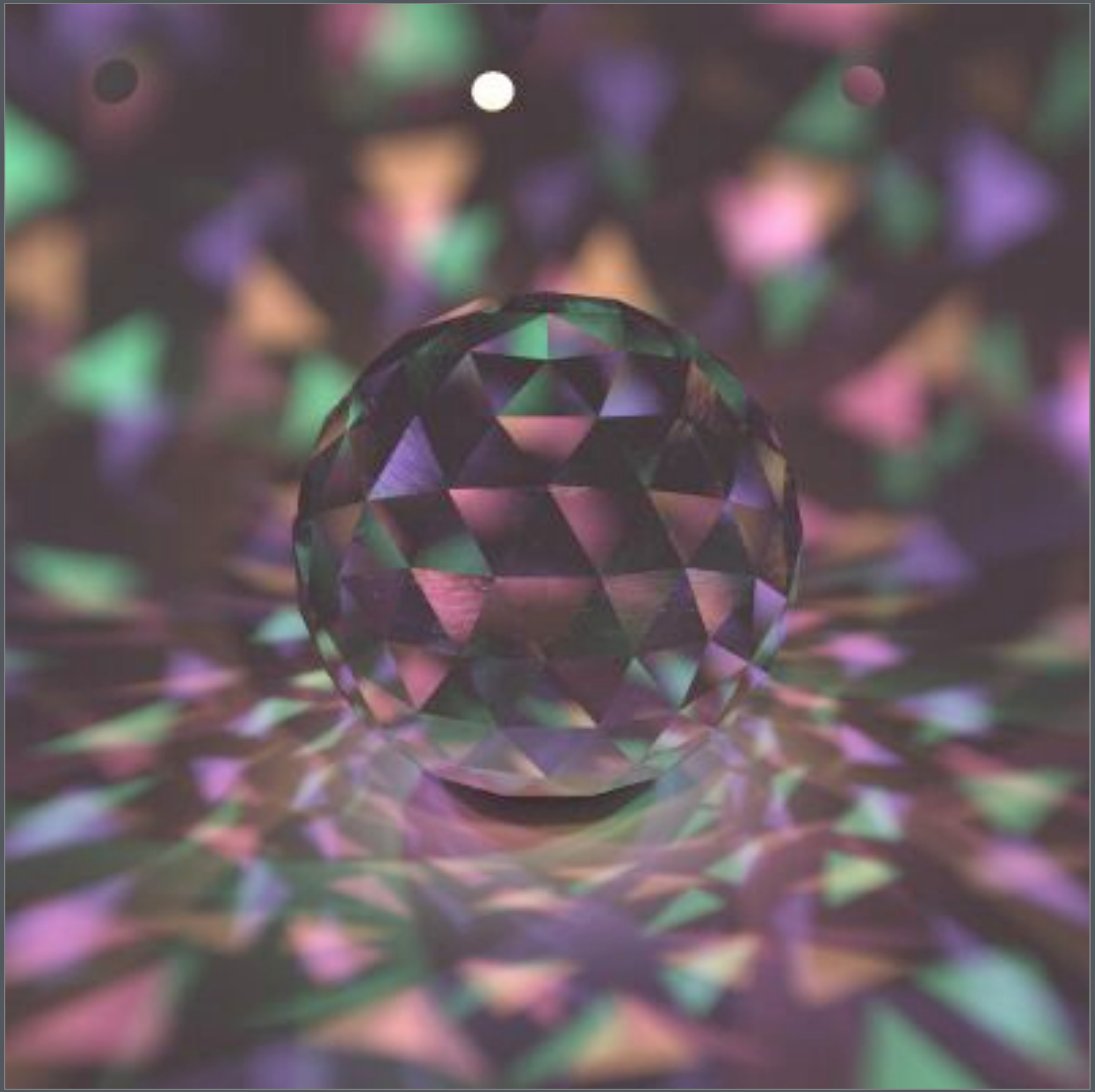
The Rendering Equation; path tracing

Lafortune and Willems 1993 • Veach and Guibas 1994

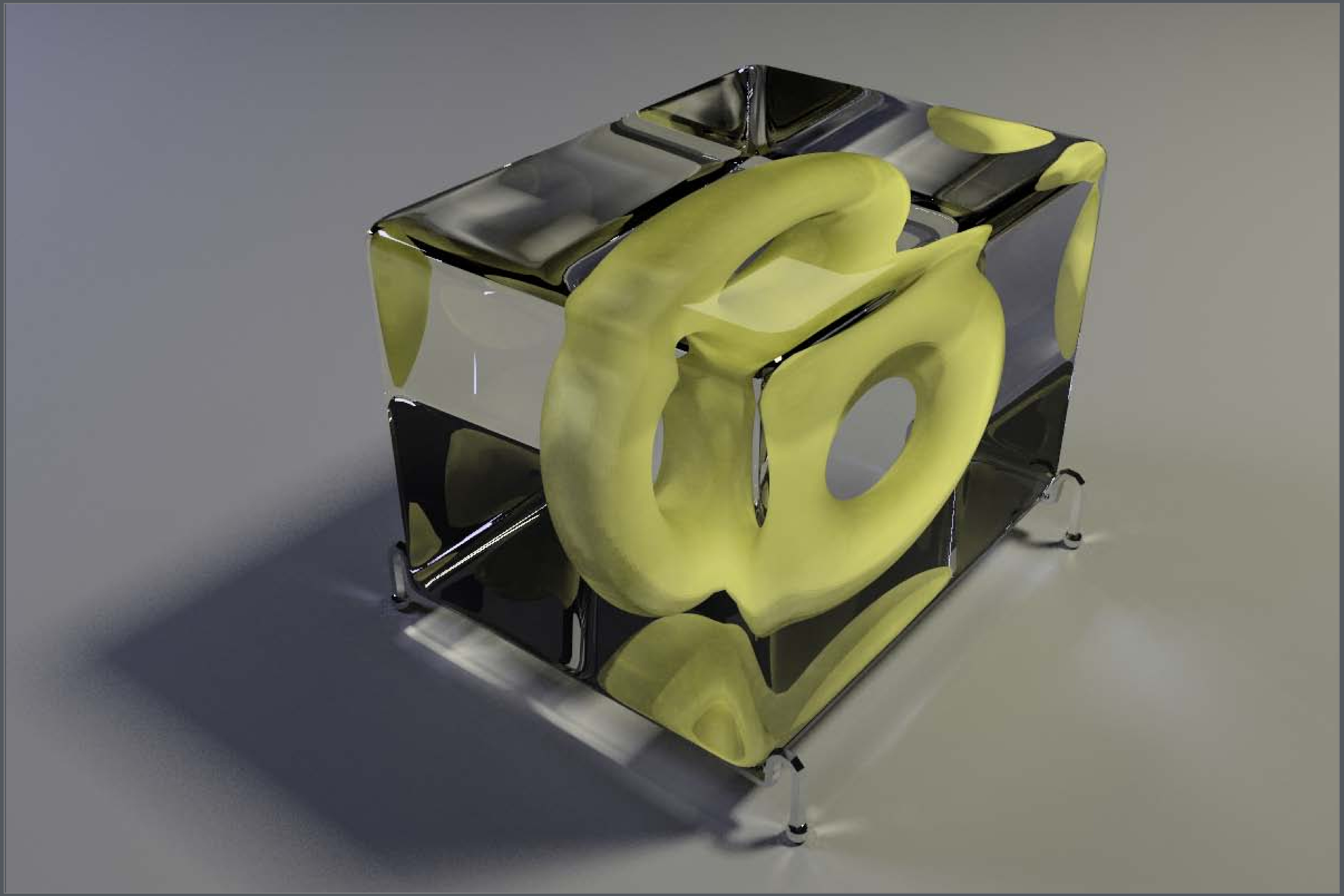
Bidirectional path tracing



Veach and Guibas 1997
Markov Chain Monte Carlo (Metropolis Light Transport)

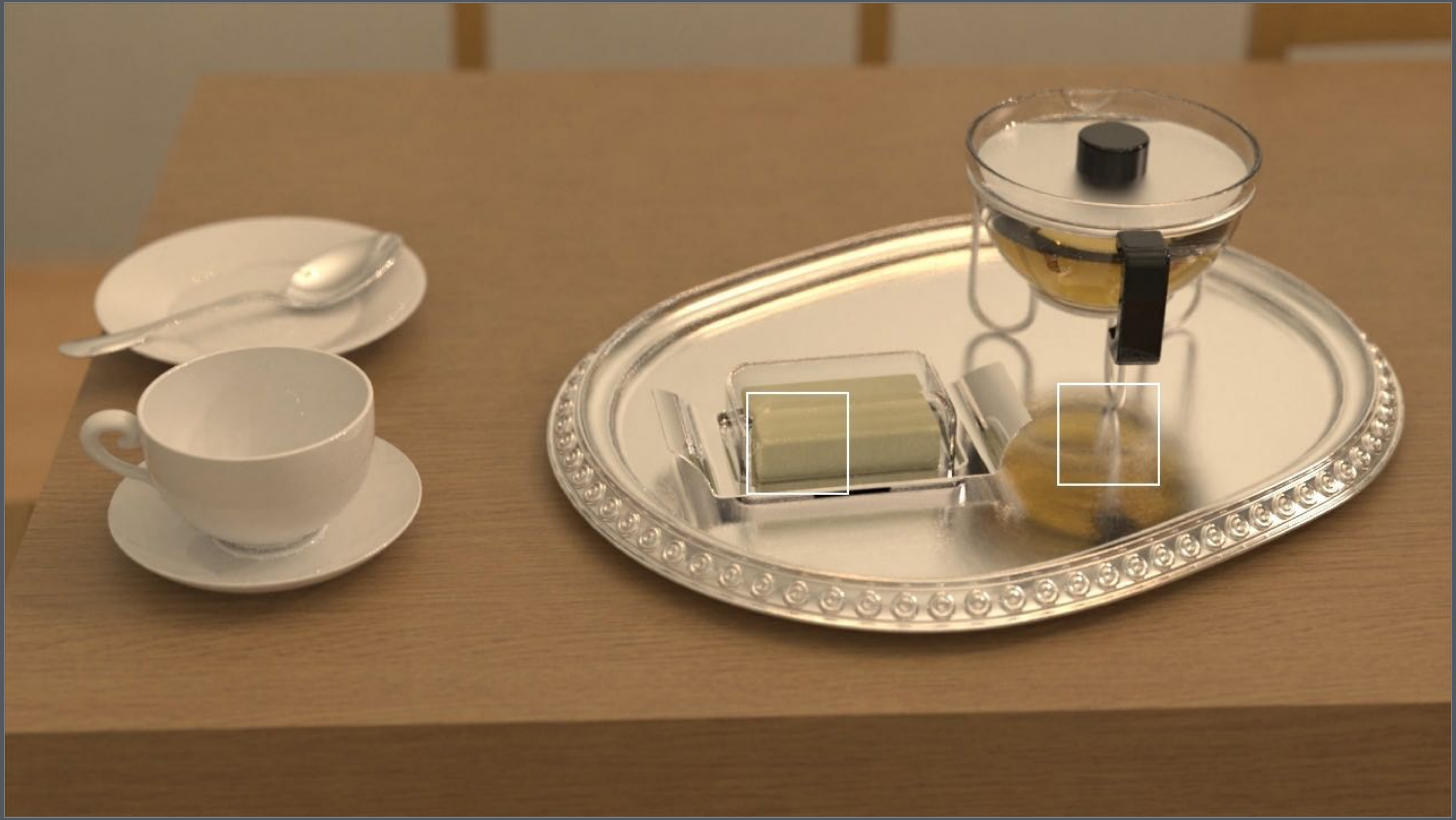


Kelemen et al. 2002
Primary sample space MCMC



Cline et al. 2005

“Energy Redistribution” with non-ergodic MCMC



Jakob & Marschner 2012
Manifold Exploration MCMC

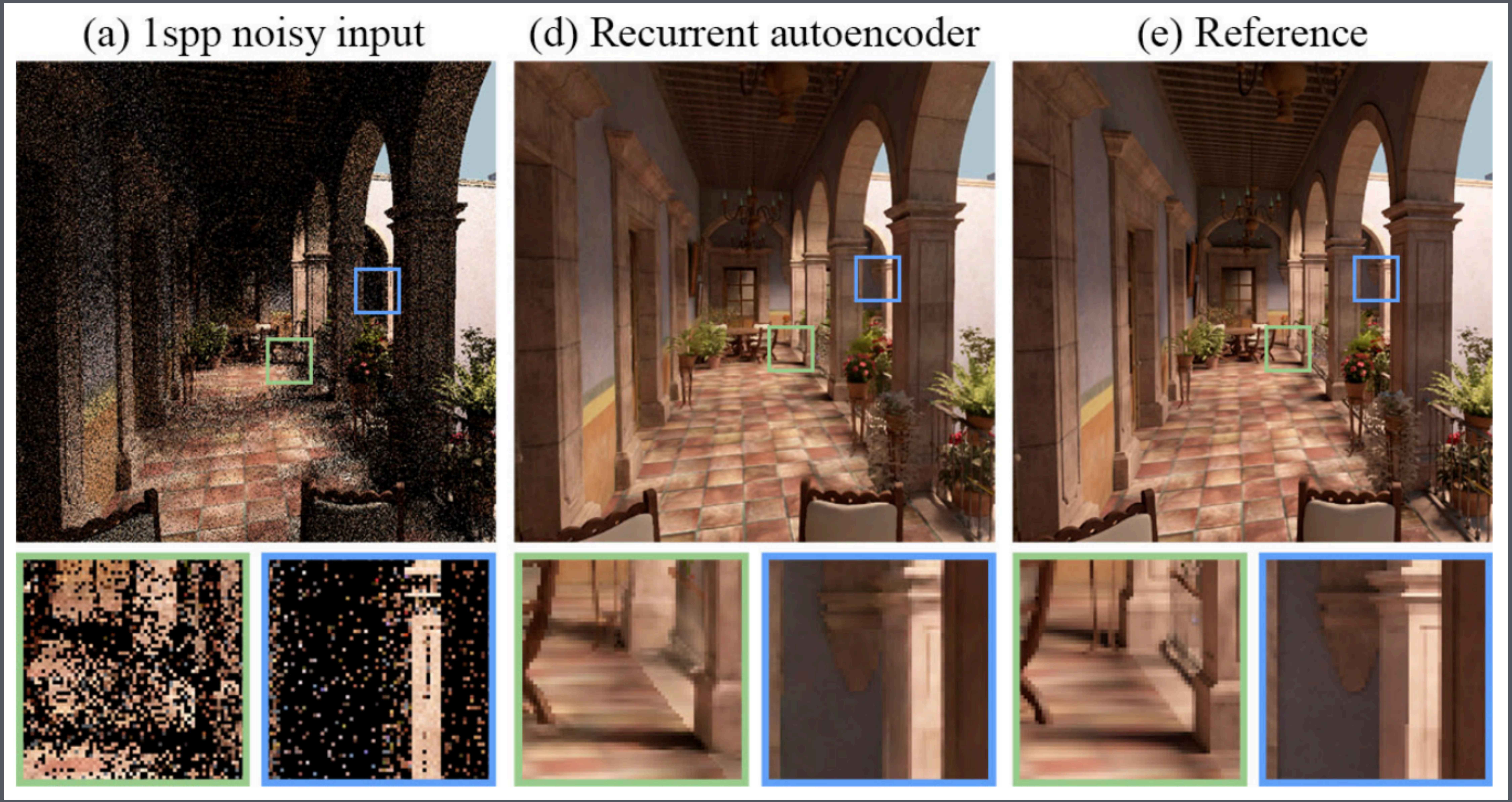
Kettunen et al. 2015

Gradient Domain Path Tracing

frame times 20-30ms

Bitterli et al. 2020

Spatiotemporal reservoir sampling



Chakravarty et al. 2017

Recurrent Denoising Autoencoder

NVIDIA Raytracing Denoisers (NRD) demo (2020) (blog post)

Real time path tracing — NVIDIA / Omniverse RTX tech demo (2020) (YouTube) (SIGGRAPH presentation)


Real time path tracing — NVIDIA RTXKit tech demo (2025) (YouTube)

About CS5630

Prerequisites

- Graphics: meshes, cameras, images, etc.
  - CS4620 or talk to instructor
- Math: integration, 3D geometry, continuous probability, multidimensional spaces
  - some mathematical maturity needed, more than specific topics
- CS: programming in C++
  - learning C++ on the fly is OK but requires extra time

Course mechanics

- Website: schedule, lecture slides, notes, readings
- CMS: Handins and grades
- Ed Discussion: Q&A

Our software framework: Nori

An educational renderer

- a simplified model of real rendering software
- same basic architecture but with complications needed for industrial strength left out
- Nori originated as the class framework built by Wenzel Jakob for CS 6630 in 2015
- now he is teaching graphics at EPFL and we borrow the improved version back from him :)

A series of assignments

- Nori starts with just the basic infrastructure of a renderer but no rendering algorithms
- you implement Monte Carlo methods, reflectance models, and a sequence of integrators
- end result can render full global illumination for simple scenes

Rendering Competition

A tradition in many realistic rendering classes

- started at Stanford in the 90s; now also at Dartmouth, UCSD, EPFL, ETH, Cornell, …

Start with a vision

- some kind of art or photograph that inspires you
- a beautiful real world phenomenon you want to render

Technical part

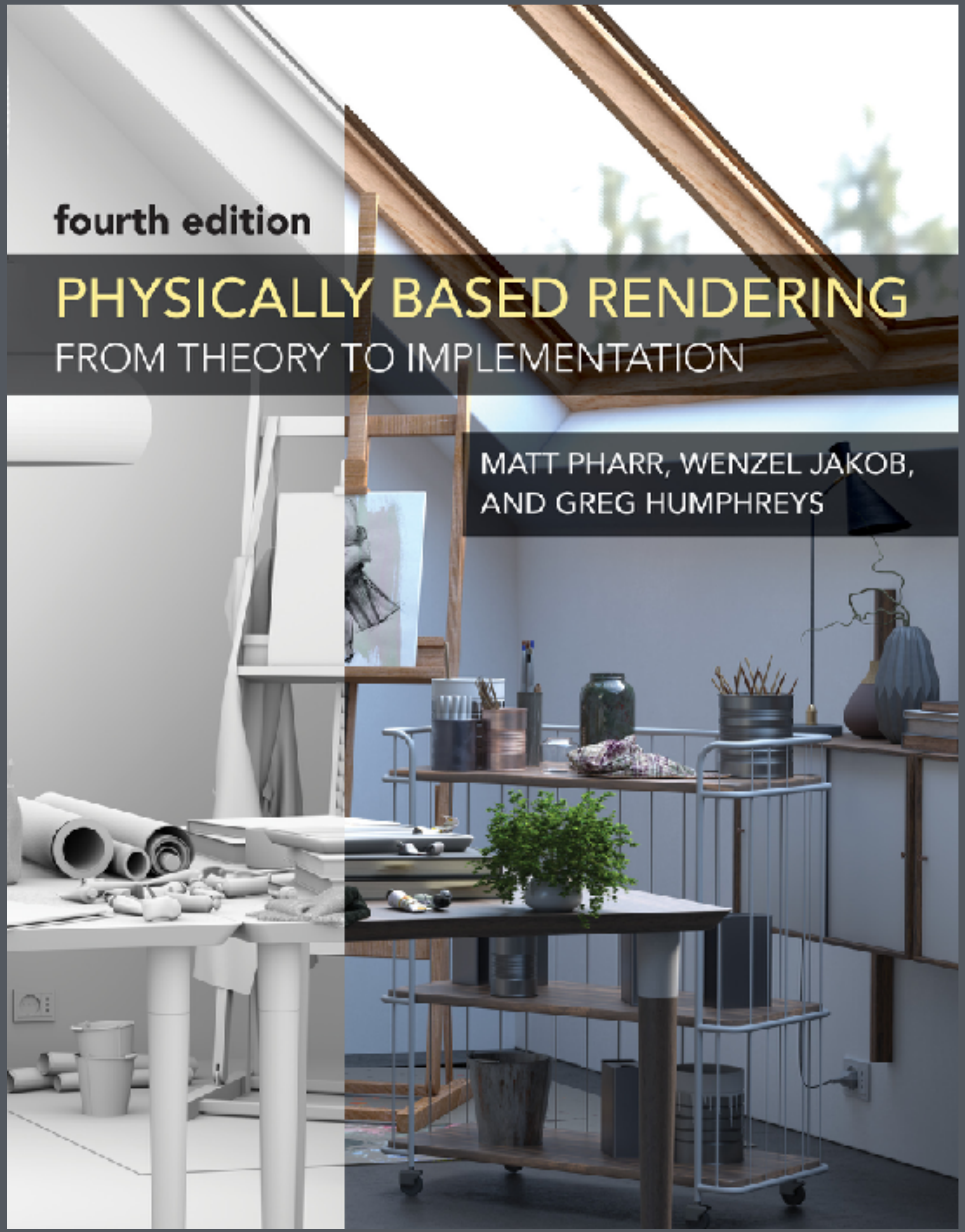
- you add capabilities to your renderer needed to achieve your vision

Artistic part

- use your program to create a great image

Competition at the end, judged by external experts

Textbooks



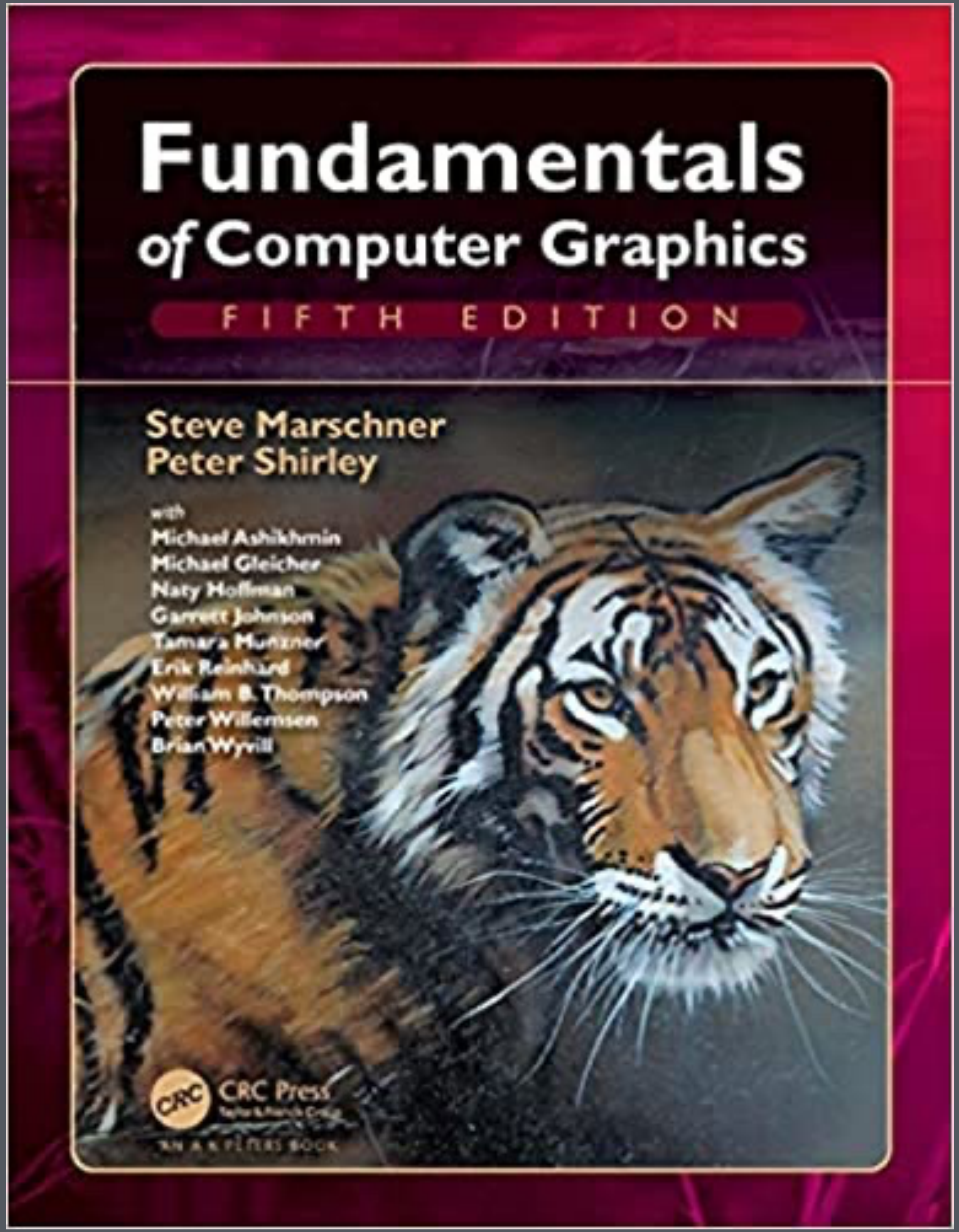
Physically-Based Rendering

- Pharr, Jakob, Humphries
- Oscar winning book on ray tracing
- available via pbr-book.org

Fundamentals of CG

- Marschner, Shirley, et al.
- for basic graphics topics

Various other readings



Grading

Course breakdown

- roughly 60% from assignments
- roughly 40% from final project
- some extra credit available

Grading

- Principle: you prove to us your code works and you understand why
  - written project report
  - interactive discussions

Generative AI

Use AI in ways that help you learn

- do: get it to help explain things (but employ skepticism!)
- do: use it to help learn about tools and techniques (especially for final project)
- do: use it to help solve C++ problems (it's seen this stuff before…)

Avoid using AI in ways that prevent learning

- don't: use its output without studying and understanding it
- don't: just have it solve the assignment and go party

We attempt to assess learning rather than only completion

- part of the grade will be based on how well your program works
- part of the grade will be based on how well you can explain your program

Academic Integrity

Don't copy code from anywhere without careful attribution

- AI generated code included
- when in doubt, just atttribute!

Collaboration

- do: help each other out, dispense advice, chat about design, help track down memory bugs
- don't: tell other students exactly what to put in their code
- don't: paste someone else's code into yours

Always cite sources of code and ideas

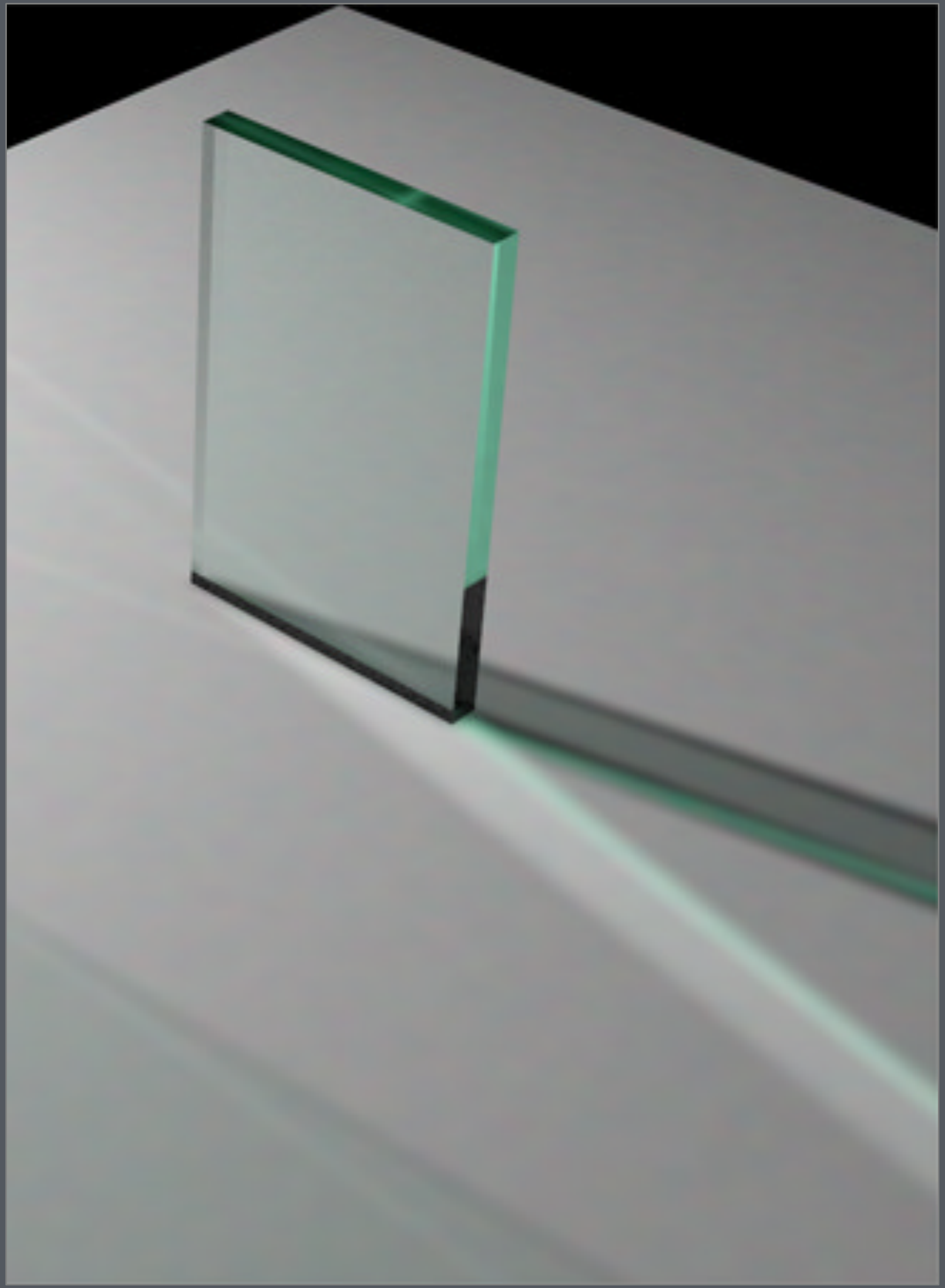
- if you tell us what is going on, there is never any AI problem

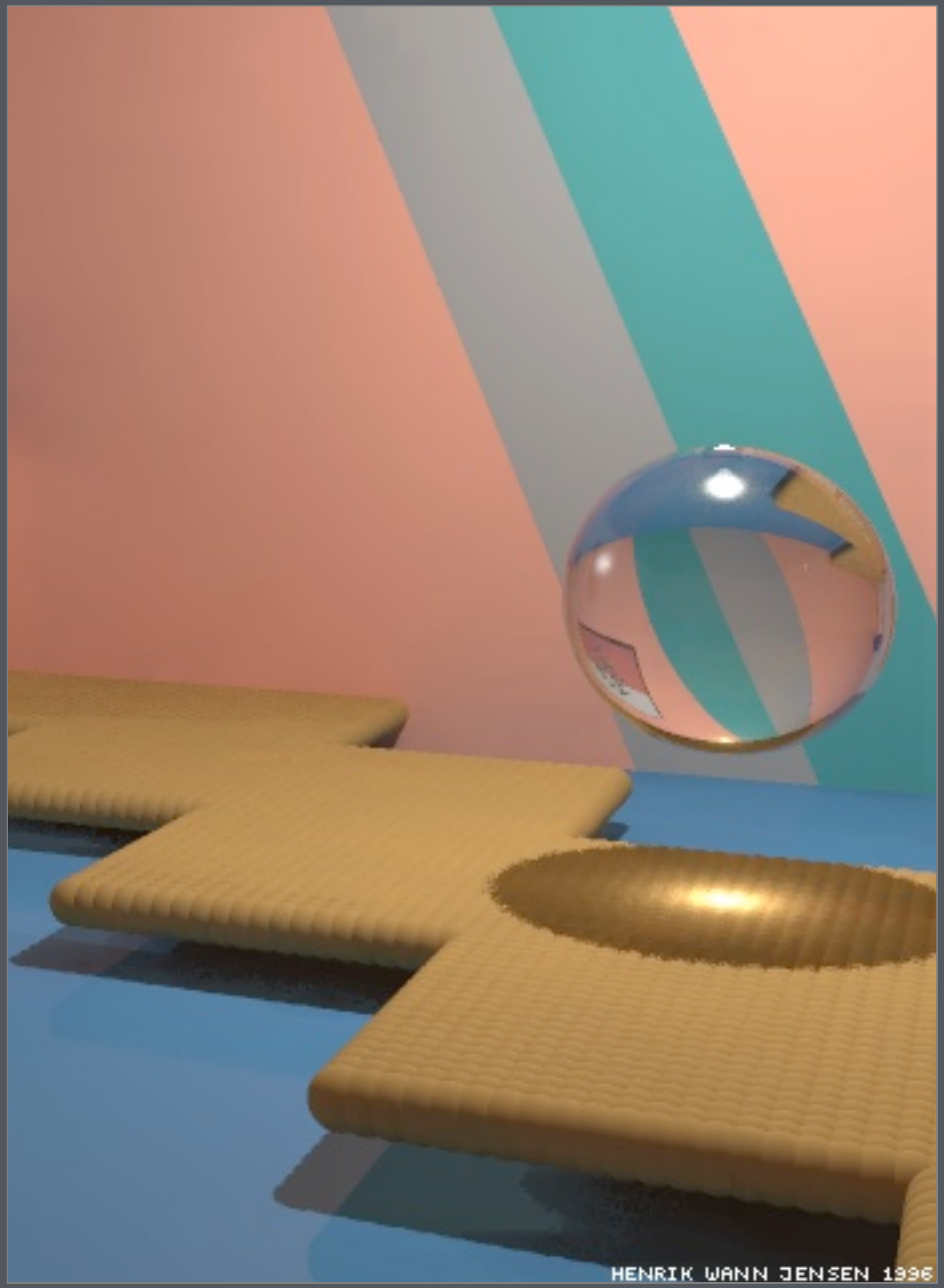
Nori assignment 0

Advanced topics

(for later in the semester)

Two-Pass Methods





Walter et al. 1997 • Jensen 1996

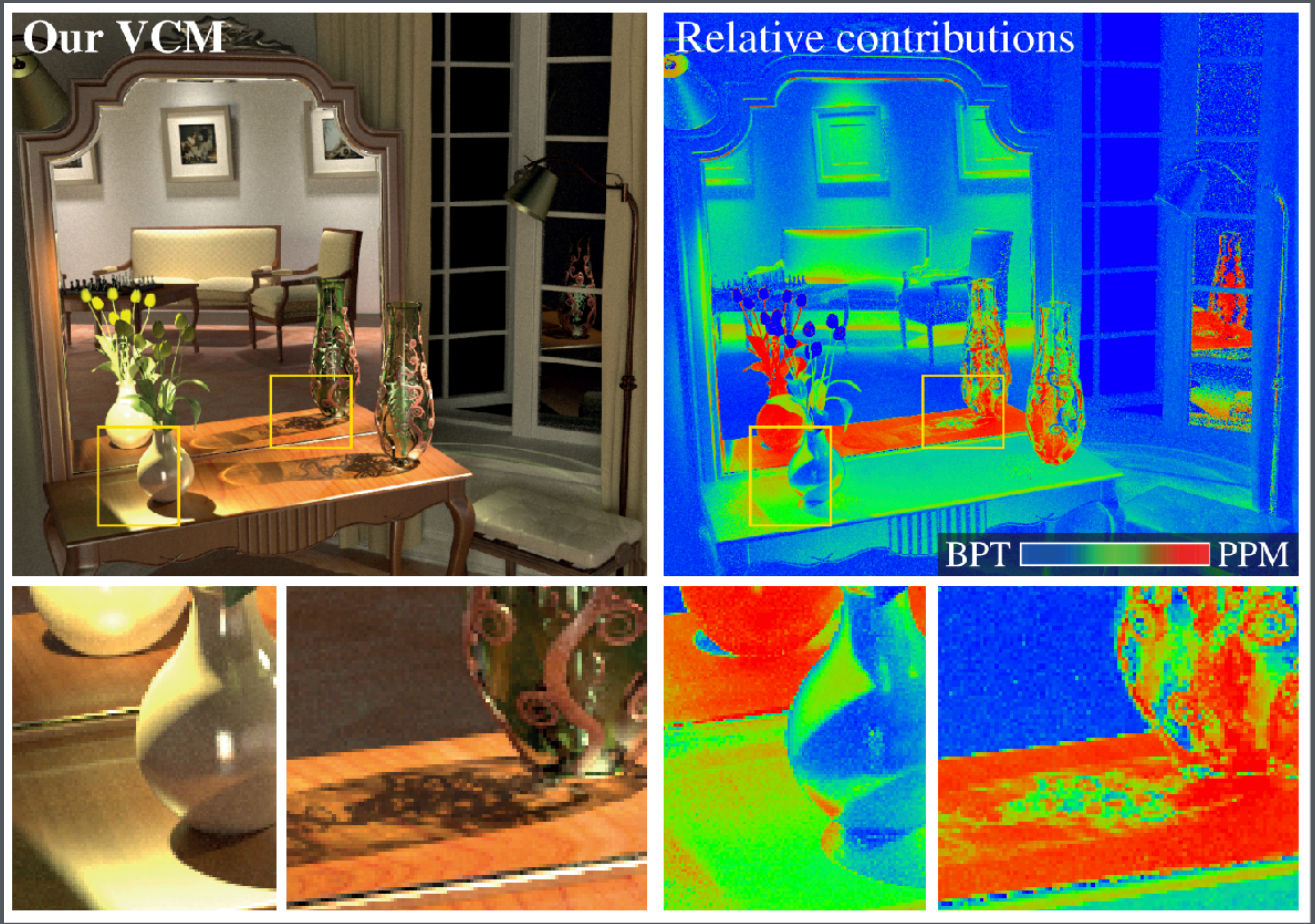
Density estimation (Photon Mapping)

Henrik Wann Jensen

Keller 1997

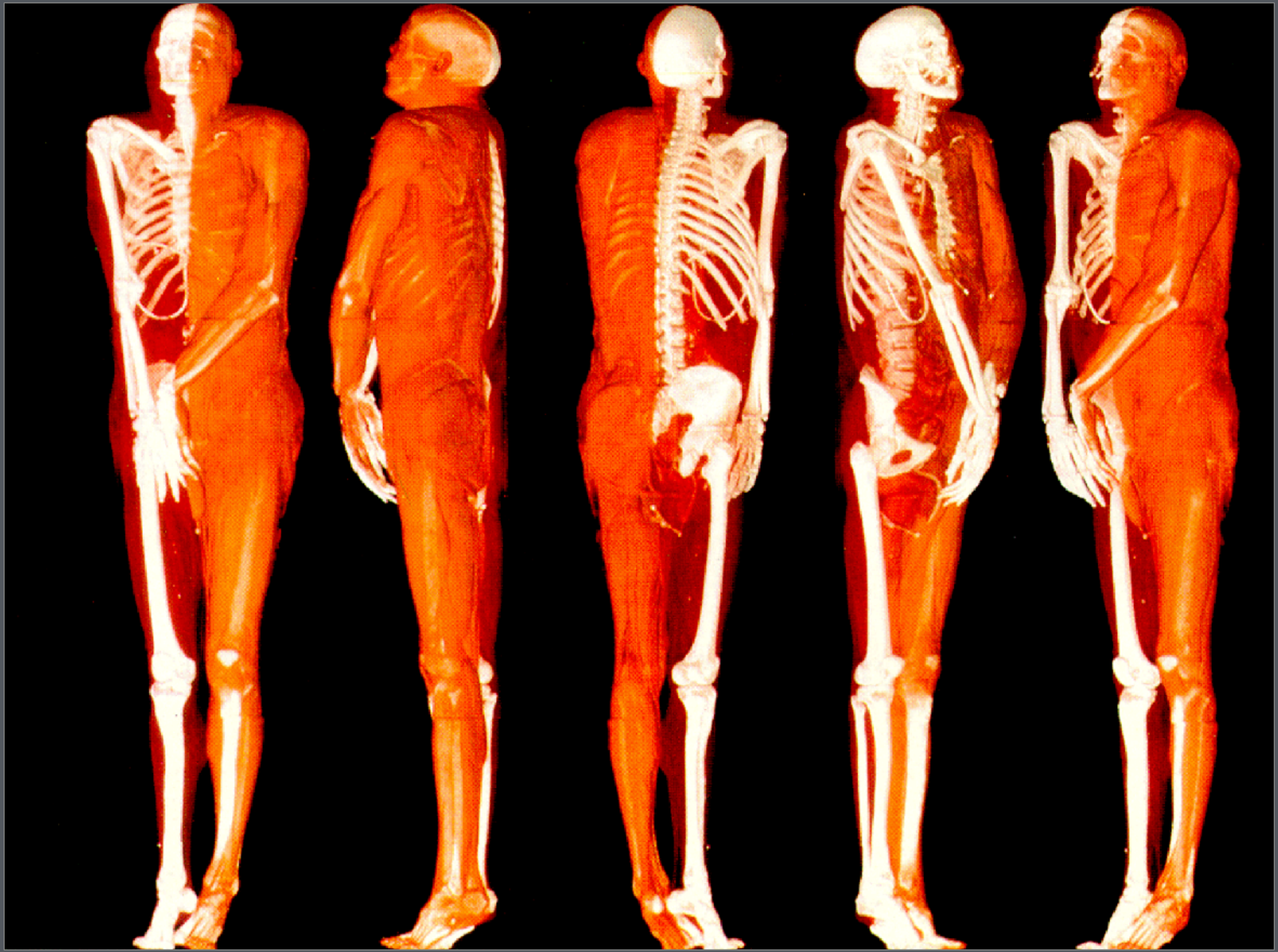
Virtual point lights (Instant Radiosity)

Walter et al. 2005
LightCuts



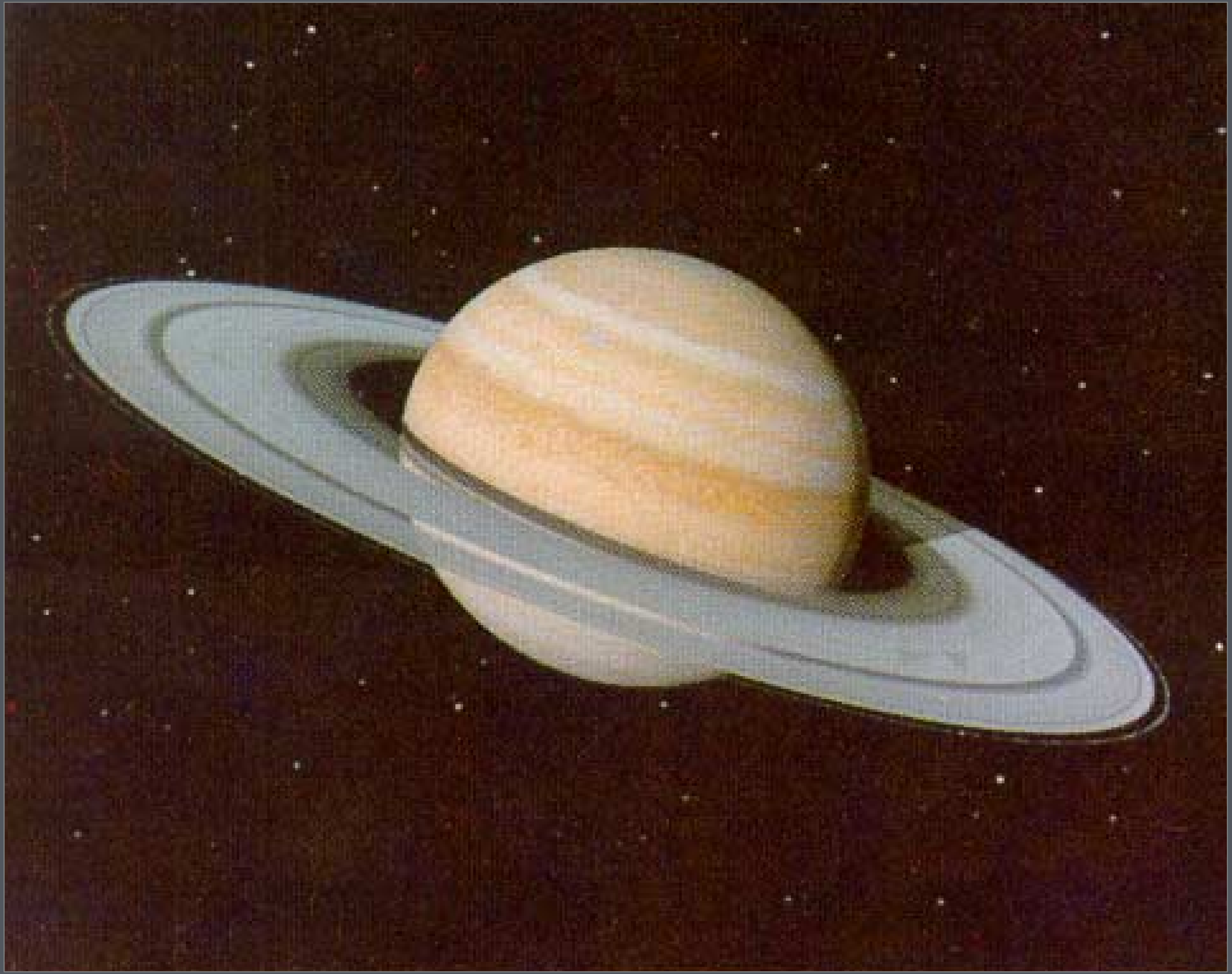
Georgiev et al. 2013
Vertex Connection and Merging

Radiative Transport



Drebin et al. 1988

Direct volume rendering



Blinn 1982

Volume scattering

(this image is later)

Jensen and Christensen 1998

Volumetric photon mapping

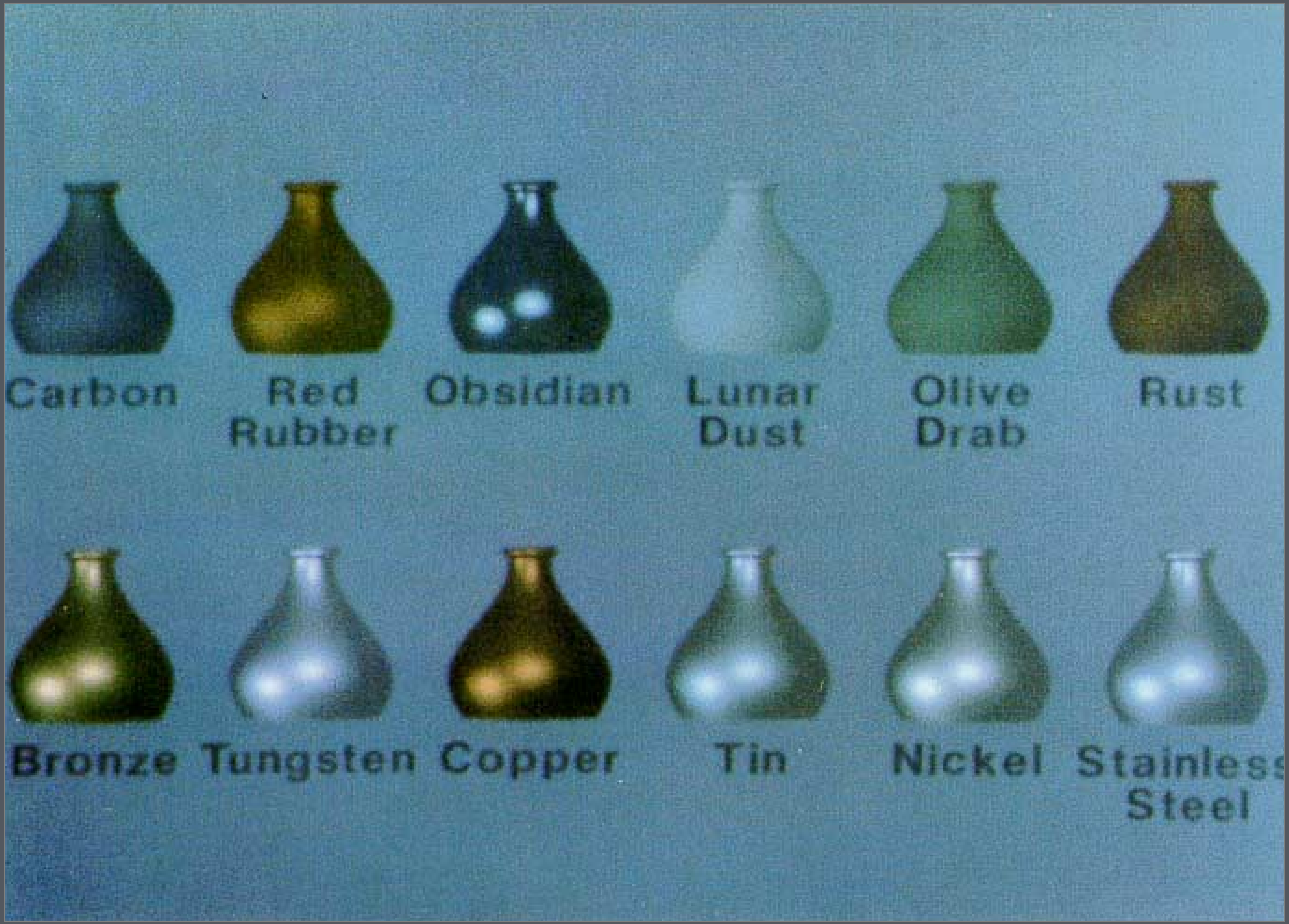
Jarosz et al. 2008
Beam Radiance Estimate

Křivánek et al. 2014
Unifying Points, Beams, and Paths

Pauly et al. 2000

Metropolis in volumes

Scattering Models

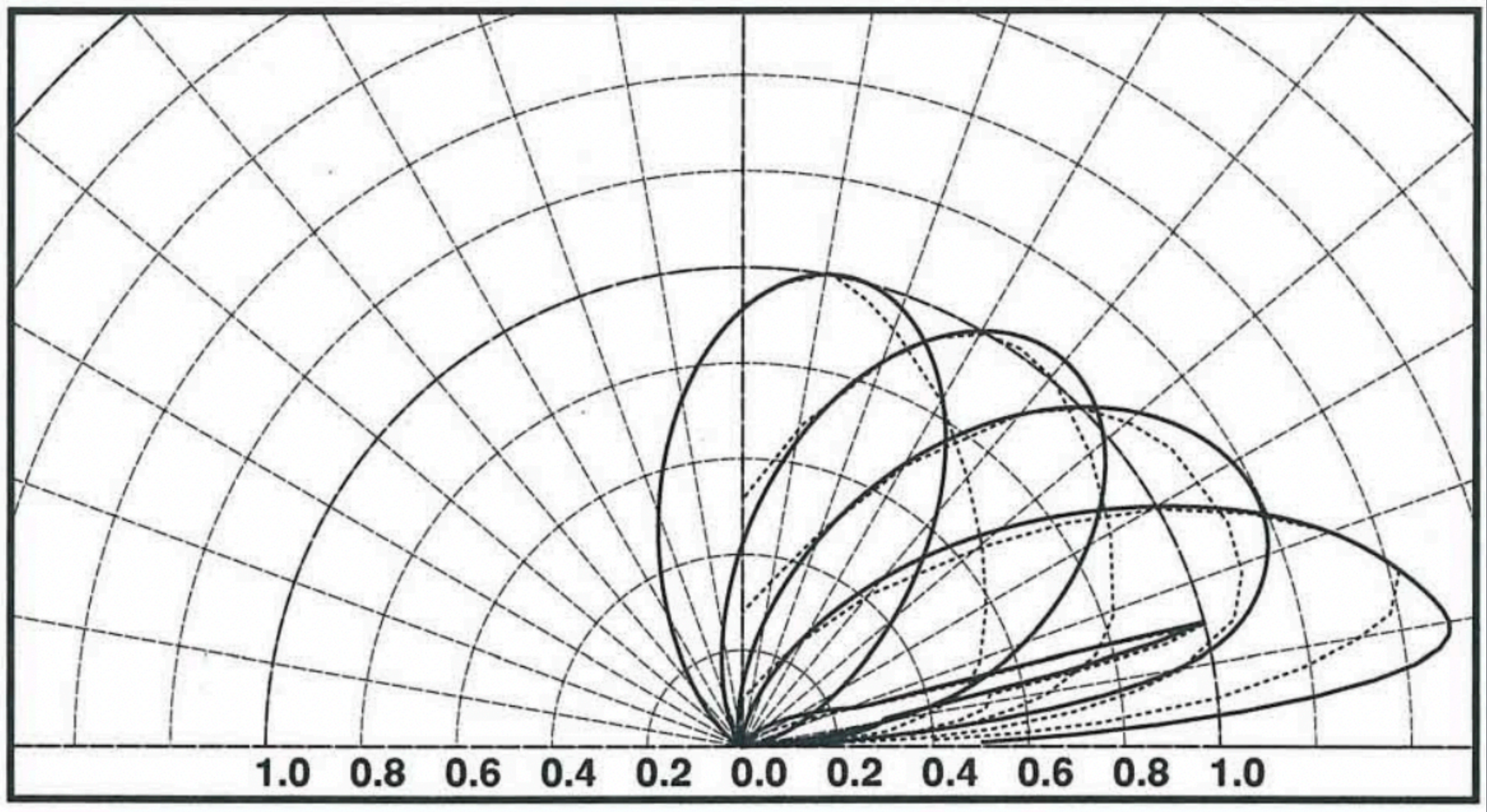


Cook and Torrance 1981

Microfacet reflection models

Walter et al. 2007

Microfacet transmission model



Xiao D. He et al. 1991

Comprehensive physical (wave) model for light reflection

Stam 1999

Fourier-based diffraction model

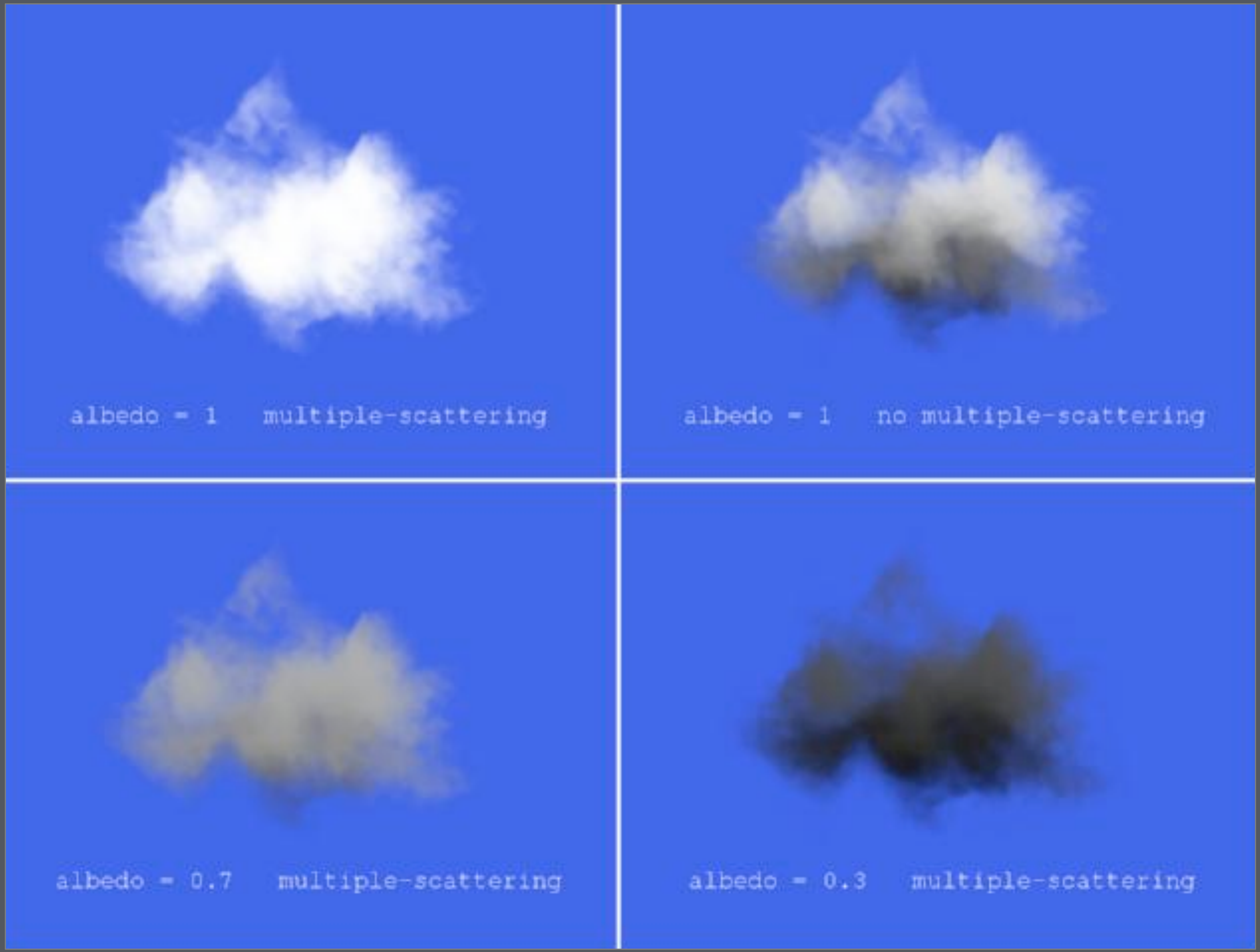
Belcour et al. 2017
Microfacet iridescence model

Jakob et al. 2014

Layered surface model

Jakob et al. 2010
Anisotropic volume media

Diffusion and Translucency



Stam 1995

Diffusion for light transport

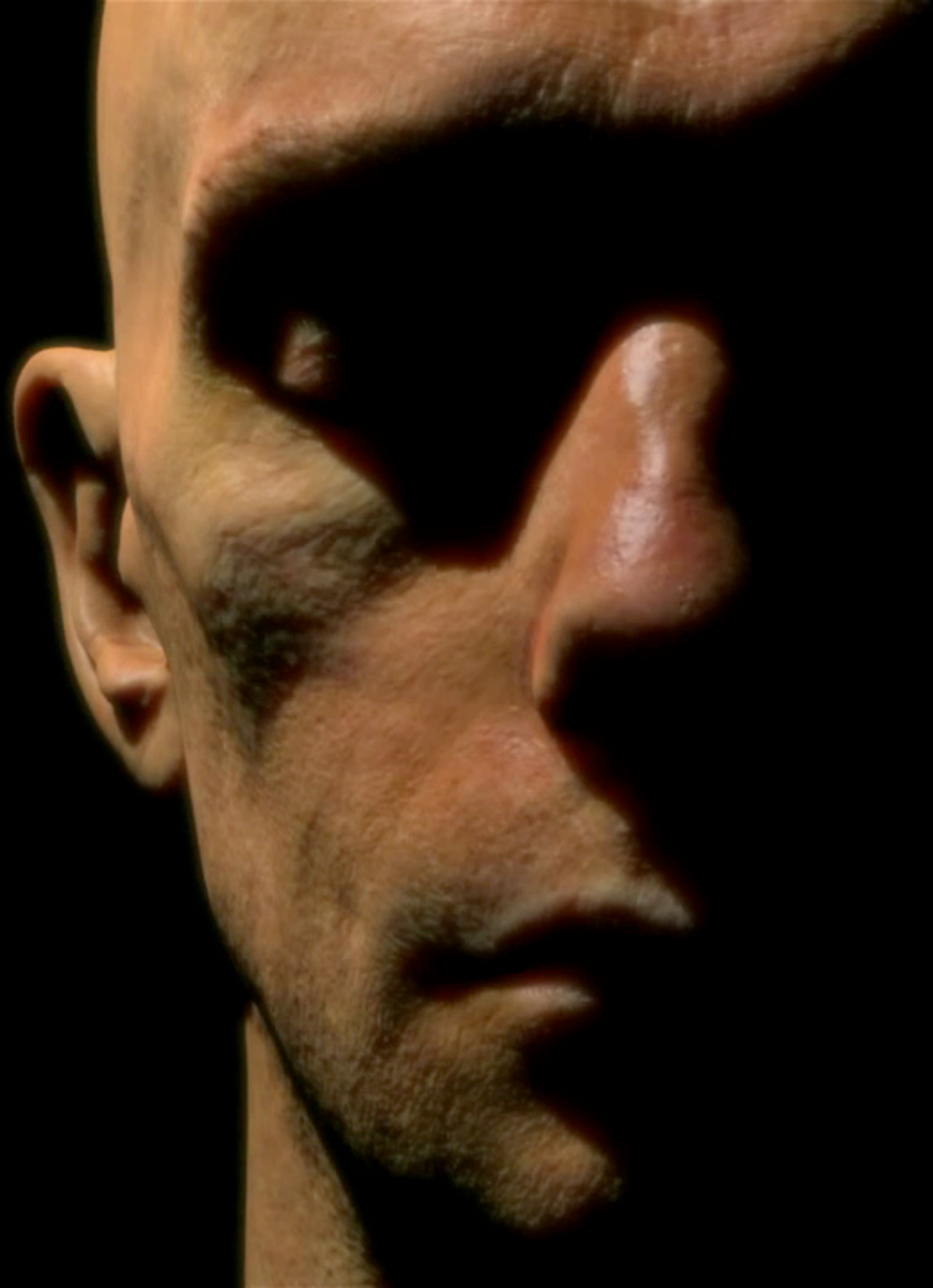

[Niniane Wang]

Jensen, Marschner, Levoy, and Hanrahan 2001

Subsurface scattering

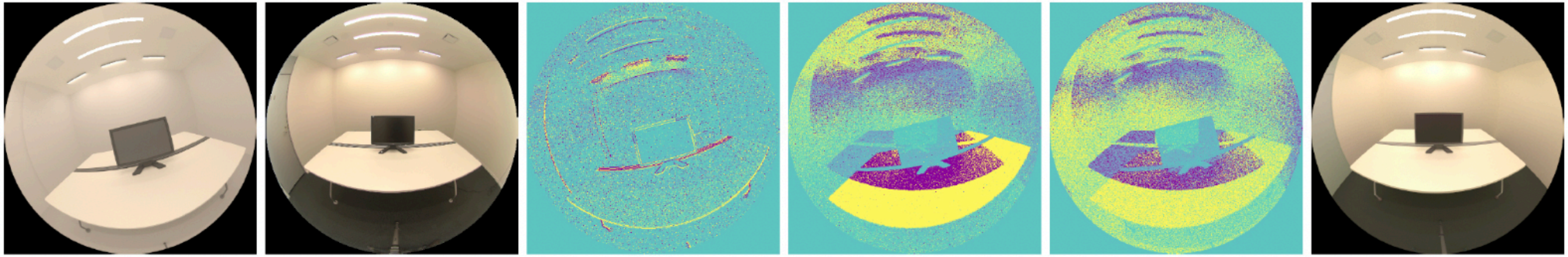
d’Eon and Irving 2011

Advanced diffusion models

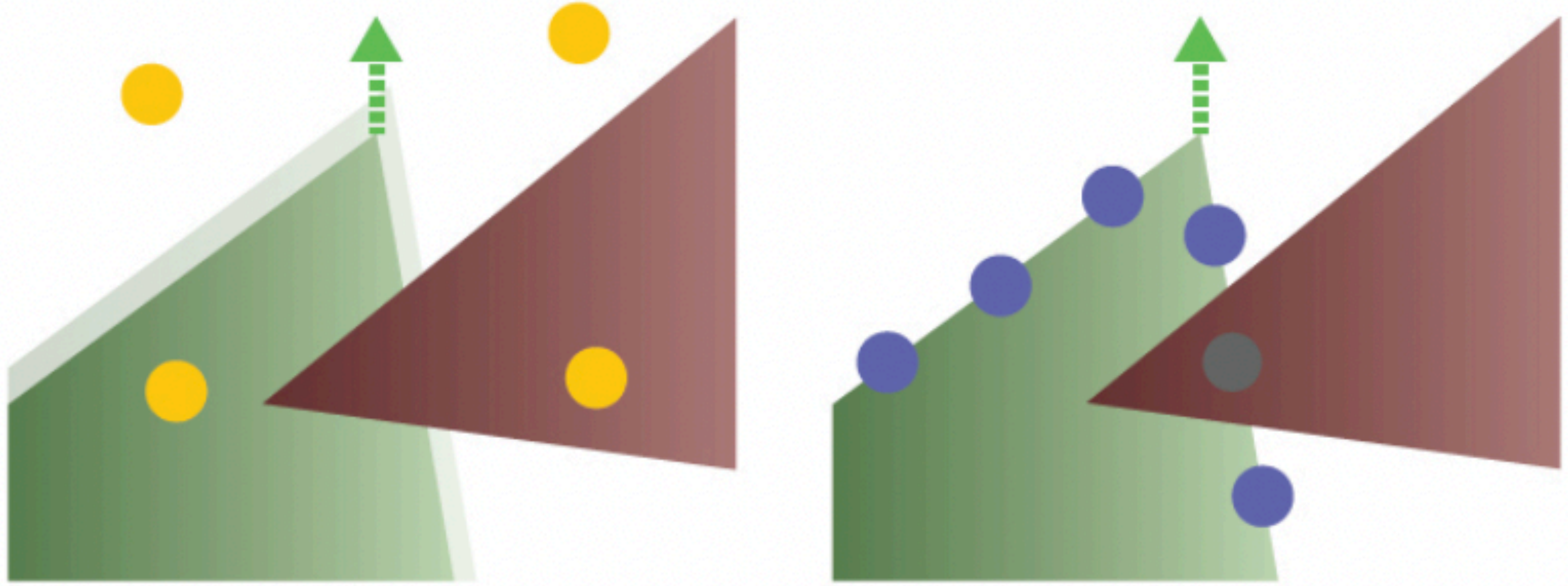


NVIDIA

Differentiable rendering



(a) initial guess (b) real photograph (c) camera gradient (per-pixel contribution) (d) table albedo gradient (per-pixel contribution) (e) light gradient (per-pixel contribution) (f) our fitted result



(a) area sampling (b) edge sampling

Tzu-Mao Li et al. 2018

Differentiable ray tracing

Original | **Derivative** with respect to sun location

Cheng Zhang et al. 2020

Path-space differentiable rendering

Nimier-David et al. 2020

Radiative backpropagation