

Path Tracing One Day Class

KAIST Visual Computing Laboratory

Ph. D. Shinyoung Yi (이신영)

See also...



shinyoung-yi / lecture-rendering-mitsuba

Type to search

[Code](#) [Issues](#) [Pull requests](#) [Discussions](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#)

lecture-rendering-mitsuba Public

[Pin](#) [Unwatch](#) 1

[main](#) [Branch](#) [Tags](#)

[Go to file](#) [t](#) [+](#) [Code](#)

shinyoung-yi	minor?	4880438 · last week	18 Commits
advanced	hw2	2 years ago	
figsrc	minor?	last week	
lecturenote	update slides & new lecture note	9 months ago	
scene	Bug fix & update code to be compatible with Mitsuba 3.5.2	9 months ago	
slides	update Lecture 01. radiometry	9 months ago	
.gitignore	Update for Mitsuba 3.6.0	3 months ago	
README.md	Update for Mitsuba 3.6.0	3 months ago	
README_ko.md	Update for Mitsuba 3.6.0	3 months ago	
hw1_radiometry.ipynb	Update for Mitsuba 3.6.0	3 months ago	
hw2_probability.ipynb	Update for Mitsuba 3.6.0	3 months ago	
hw3_directillumination.ipynb	Update for Mitsuba 3.6.0	3 months ago	
hw4_pathtracing.ipynb	Update for Mitsuba 3.6.0	3 months ago	
tutorial1-1 (optional).ipynb	Update for Mitsuba 3.6.0	3 months ago	
tutorial1_Sampler.ipynb	Update for Mitsuba 3.6.0	3 months ago	
tutorial2_ray_intersection.ipynb	Update for Mitsuba 3.6.0	3 months ago	
tutorial3_BSDF.ipynb	Update for Mitsuba 3.6.0	3 months ago	
tutorial4_Emitter.ipynb	Update for Mitsuba 3.6.0	3 months ago	
util.py	Update for Mitsuba 3.6.0	3 months ago	

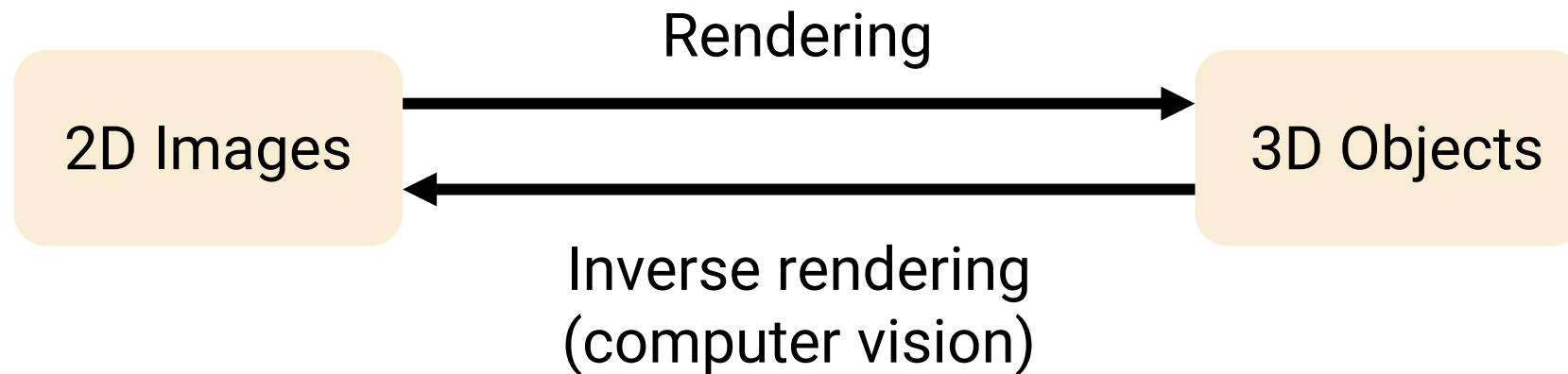
<https://github.com/shinyoung-yi/lecture-rendering-mitsuba>

- Full version slides for each chapter:
 - 1. Radiometry and Light Transport
 - 2. Probability and Statistical Inference
 - 3. Path Tracing
- Tutorial for Mitsuba 3 API
- Python skeleton codes for homework

Why Should We Study Ray Tracing?



“I am only interested in computer vision”



- Generate GT dataset for training
- Understand exact problem definitions
- Understanding **GT** forward methods

global illumination, participating media, etc.

Why Should We Study Ray Tracing?



**“I only want to *use* rendering,
not to understand”**



Why Should We Study Ray Tracing?



**“I only want to *use* rendering,
not to understand”**

Reconstructing Translucent Objects using Differentiable Rendering

Xi Deng
Cornell University
Ithaca, NY, USA
xd93@cornell.edu Fujun Luan
Adobe Research
San Jose, CA, USA
fluan@adobe.com Bruce Walter
Cornell University
Ithaca, NY, USA
bruce.walter@cornell.edu Kavita Bala
Cornell University
Ithaca, NY, USA
kavita@cs.cornell.edu Steve M. Dror
Cornell University
Ithaca, NY, USA
smr@cs.cornell.edu

SIGGRAPH '22 Conference Proceedings, August 7–11, 2022, Vancouver, BC, Canada

For the surface reflectance, we use a rough dielectric model with GGX distribution (Walter et al. 2007), which is parameterized by a roughness β and the index of refraction η of the surface. The parameters of BSSRDF and BRDF can be represented using texture maps on the surface or a uniform value.

Less Function. We formulate our reconstruction problem as finding a parameter vector π , which includes the vertex positions of the mesh, the surface roughness, and the subsurface albedo α and extinction coefficient σ_t , by minimizing a loss function g that describes the difference between renderings $I(\pi)$ and reference images I_{ref} :

$$\arg \min_{\pi} g(I(\pi)). \quad (5)$$

We measure the difference as a sum of squares, but instead of using the conventional L_2 loss we define the loss function as

$$g(I(\pi)) = (\mathbf{I}_1(\pi) - \mathbf{I}_{ref})(\mathbf{I}_2(\pi) - \mathbf{I}_{ref}), \quad (6)$$

where $\mathbf{I}_1(\pi)$ and $\mathbf{I}_2(\pi)$ are two statistically independent Monte Carlo estimates of the noise-free image $I(\pi)$. We show in Section 5 that our loss function effectively provides the required unbiased estimate.

Differentiable Renderer. Efficiently solving (5) requires the gradient of g with respect to π , computed via the derivatives of pixel values with respect to each parameter. Previous work (Zhang et al. 2020) shows how to do differentiable path tracing under the assumption of area lights and surface reflection modeled by the Bidirectional Scattering Distribution Function (BSDF) without ideal specular materials using a path integral framework.

4 DIFFERENTIABLE TRANSLUCENT RENDERING

In this section we extend the path integral framework of Zhang et al. (2020) and generalize the image derivative estimates to the setting of translucent materials.

4.1 Path Integral Framework with BSSRDF

In the path integral formulation [Veach 1998] a pixel value is:

$$I = \int_{\tilde{\Omega}} f(\tilde{x}) d\mu(\tilde{x}). \quad (7)$$

where $\tilde{x} = y_0 \cdot \dots \cdot y_n$ is the complete light path with its ends y_0 and y_n at the camera and the light source respectively. We define two categories of path segments y_{i-1}, y_i : subsurface segments going through translucent material and vacuum segments that pass through empty space. We refer to the two vertices on a subsurface segment as subsurface transmission vertices and other vertices as scattering vertices.

A measure on a path with length n is usually written as a product of area measures $\mu(\tilde{x}) = \prod_{i=0}^{n-1} d\mu(y_i)$ and all the paths with length n form an integration domain Ω_n ; this domain depends on π so we denote it as $\Omega_n(\pi)$. The space of all possible lengths forms the path space $\Omega(\pi) = \cup_{n=1}^{\infty} \Omega_n(\pi)$.

The integrand of (7) is the contribution of a single light path,

$$f(\tilde{x}) = W_c(y_0, y_1) C(\tilde{x}) G(\tilde{x}) I_c(y_n, y_{n-1}), \quad (8)$$

Xi Deng, Fujun Luan, Bruce Walter, Kavita Bala, and Steve M. Dror. 2022. Reconstructing Translucent Objects using Differentiable Rendering. In *Special Interest Group on Computer Graphics and Interactive Techniques (SIGGRAPH '22 Conference Proceedings, August 7–11, 2022, Vancouver, BC, Canada*). ACM, New York, NY, USA, 1–10. DOI: <https://doi.org/10.1145/3528233.3530714>

Permission to make digital or hard copies of all or part of this work for personal use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyright © 2022, Association for Computing Machinery (ACM). Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 978-1-4503-9372-0/22/08...\$15.00

Reconstructing Translucent Objects using Differentiable Rendering

Xi Deng, Fujun Luan, Bruce Walter, Kavita Bala, and Steve M. Dror

SIGGRAPH '22 Conference Proceedings, August 7–11, 2022, Vancouver, BC, Canada

indirect illumination neglected, we specialize Eq. (16) to

$$I = \int_{\tilde{\Omega}_1} \hat{f}(\tilde{p}) d\mu(\tilde{p}) + \int_{\tilde{\Omega}_2} \hat{f}(\tilde{p}) d\mu(\tilde{p}) \quad (21)$$

$$= \int_M \int_M \hat{f}(p_0, p_1, p_2) dA(p_1) dA(p_2) + \int_M \int_M \hat{f}(p_0, p_1, p_2) dA(p_1) dA(p_2) \quad (22)$$

where the contribution $\hat{f}(p_0, p_1, p_2)$ from a path with a subsurface segment in the middle is

$$W_c V(y_0, y_1) S_w(y_1) / (p_1) R_d(y_1, y_2) S_w(y_2) J(p_2) V(y_2, y_3) L_c \quad (23)$$

and similarly, the contribution from the path consisting of two vacuum segments is $\hat{f}(p_0, p_2) = W_c V(y_0, y_1) J(y_1) \rho_s(y_0, y_1) V(y_1, y_2) L_c$.

Then differentiating the BSSRDF part of Eq. (22) with respect to scene parameters π using Eq. (20) gives us

$$\partial I_s / \partial \pi = \frac{d}{d\pi} \int_{\tilde{\Omega}} \hat{f}(\tilde{p}) d\mu(\tilde{p}) \quad (24)$$

$$= \int_M \int_M \frac{\partial \hat{f}}{\partial \pi}(p_0, p_1, p_2) \frac{\partial C(\tilde{p})}{\partial \pi} \frac{\partial dA(p_1)}{\partial \pi} dA(p_1) dA(p_2) \quad (25)$$

$$+ \int_M \int_M \int_{\tilde{\Omega}} \hat{f}(p_0, p_1, p_2) v_B(p_2) dA(p_1) dA(p_2) \quad (26)$$

$$+ \int_M \int_M \int_{\tilde{\Omega}} \Delta \hat{f}(p_0, p_1, p_2) v_B(p_2) dA(p_1) dA(p_2), \quad (27)$$

boundary on p_2

where $v_B(p)$ is the velocity (gradient w.r.t. x) of the visibility boundary in the direction of the normal to $dA(p)$, shown in Fig. 3.

4.4 Monte Carlo Estimation of the Derivatives

The path integral and the differentiable path integral are both estimated by a Monte Carlo algorithm: $I = \frac{1}{N} \sum_{i=1}^N \frac{f(\tilde{x}_i)}{p(\tilde{x}_i)}$, and in our problem setup, particularly,

$$(I_s) = \frac{1}{N} \sum_{i=1}^N \frac{f(p_0, p_1, p_2)}{p(y_1) p(y_2)}, \quad (I_2) = \frac{1}{N} \sum_{i=1}^N \frac{f(p_0, p_1, p_2)}{p(y_1)}, \quad (28)$$

where the paths in three segments contain one subsurface segment in the middle. Then the interior term in Eq. (25) is estimated by

$$\frac{1}{N} \sum_{i=1}^N \frac{\partial f(p_0, p_1, p_2)}{\partial \pi}. \quad (29)$$

where the path is sampled just as in conventional path tracing.

To estimate the integrals in Eq. (27) and Eq. (26), paths are sampled on the discontinuity boundaries, including the primary visibility boundary (27) which relates to silhouette edges in the image, and the secondary visibility boundary (26), which relates to edges on the mesh surface. These boundaries interact with opaque surface materials, where the discontinuity caused by the shadow edge is directly projected back to the camera, in our case this discontinuity is blurred by the BSSRDF on the surface and its motion indirectly affects the outgoing radiance at surface points near the shadow edge.

5

Why Should We Study Ray Tracing?



**“I only want to *use* rendering,
not to understand”**

- Choosing ray tracing algorithms often requires understanding them
- We sometimes need to write special-purposed ray tracers

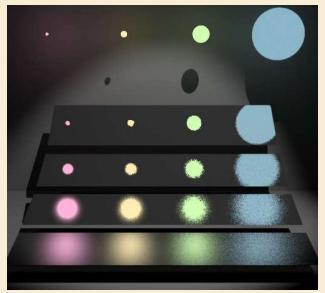
Why Should We Study Ray Tracing?



“Too hard and too many methods to study”



[Cook 1984]



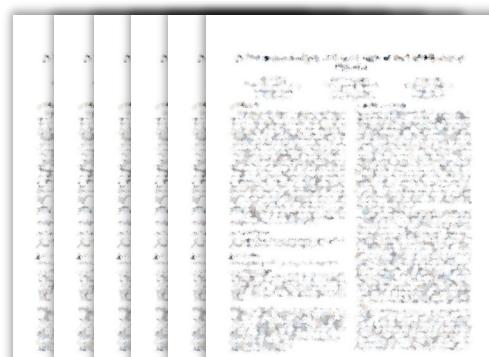
[Veach & Guibas 1995]



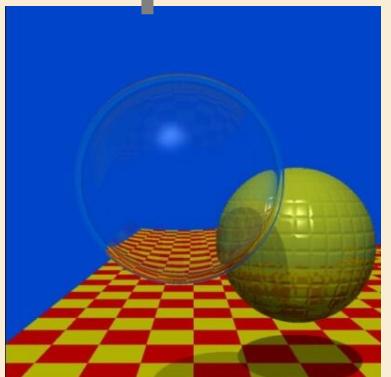
[Veach & Guibas 1995]



[Veach & Guibas 1995]



2020



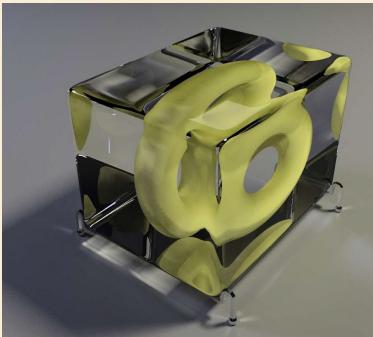
[Whitted 1980]



[Kajiya 1986]



[Veach & Guibas 1997]



[Cline et al. 2005]



Why Should We Study Ray Tracing?

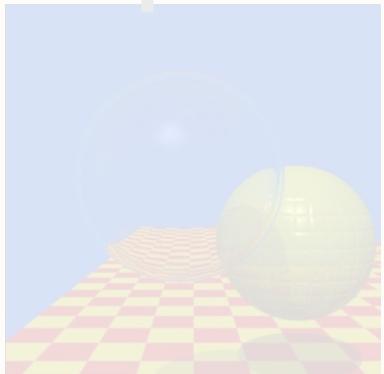


“Too hard and too many methods to study”

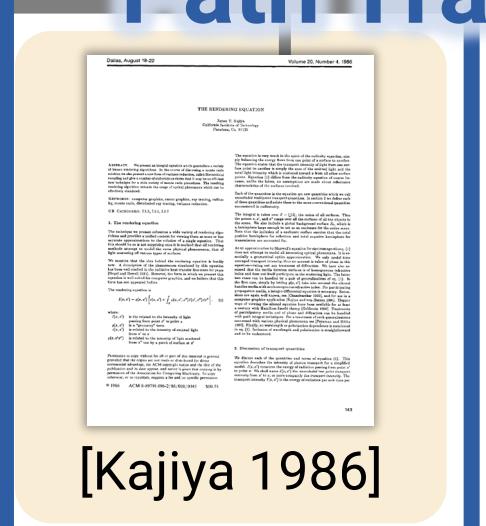
Early ray tracing
We can skip!

1980

[Cook 1984]



[Whitted 1980]



[Kajiya 1986]



[Veach &
Guibas 1995]

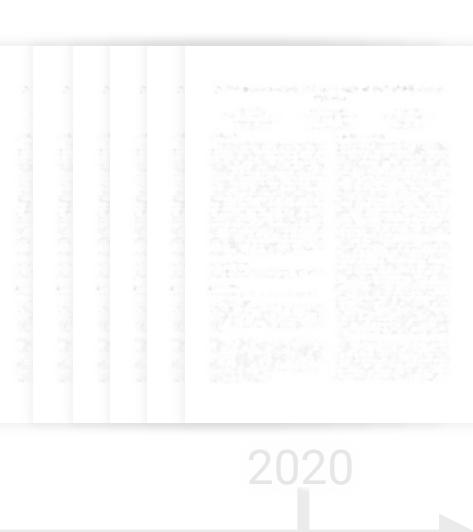
Path Tracing



[Veach &
Guibas 1995]



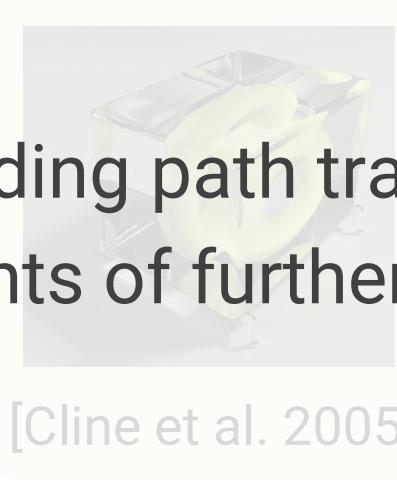
[Veach &
Guibas 1995]



2020



[Veach & Guibas
1997]



[Cline et al. 2005]

Why Path Tracing?



**Path Tracing is the most popular
standard method of ray tracing!**

Q. How popular? How standard?

Path tracing in popular software



Mitsuba 3

PHYSICALLY BASED RENDERER

Path tracer (path)

Parameter	Type	Description	Flags
max_depth	integer	Specifies the longest path depth in the generated output image (where -1 corresponds to ∞). A value of 1 will only render directly visible light sources, 2 will lead to single-bounce (direct-only) illumination, and so on. (Default: -1)	
rr_depth	integer	Specifies the path depth, at which the implementation will begin to use the <i>russian roulette</i> path termination criterion. For example, if set to 1, then path generation many randomly cease after encountering directly visible surfaces. (Default: 5)	
hide_emitters	boolean	Hide directly visible emitters. (Default: no, i.e. false)	

This integrator implements a basic path tracer and is a **good default choice** when there is no strong reason to prefer another method.

To use the path tracer appropriately, it is instructive to know roughly how it works: its main operation is to trace many light paths using *random walks* starting from the sensor. A single random walk is shown below, which entails casting a ray associated with a pixel in the output image and searching for the first visible intersection. A new direction is then chosen at the intersection, and the ray-casting step repeats over and over again (until one of several stopping criteria applies).

**"a good default choice
when there is no strong reason
to prefer another method."**



Scene

Render Engine: Cycles

Feature Set: Supported

Device: CPU

Open Shading Language

Sampling

Integrator: Path Tracing (highlighted with a red box)

Render: Branched Path Tracing 128

Viewport: Path Tracing 32

Adaptive Sampling

Integrator

Path tracing is still used for comparison

- “PT” in comparison
 - [TOG’19] Neural Importance Sampling
 - [TOG’20] Path Cuts
 - [TOG’20] Specular manifold sampling
 - [TOG’21] Hierarchical Neural Reconstruction for Path Tracing
 - [TOG’21] Vectorization
 - [TOG’21] Path Graphs

Fig. 1. We present a hierarchical path tracing framework in which a camera, light, and phone sensors to generate high-quality samples for rendering. The figure shows a comparison of different sampling strategies. The columns represent different methods: Path (baseline), Sobol (low-discrepancy), Molar (adaptive), Molt (adaptive), Ravi et al. (adaptive), Zhou et al. (adaptive), Paliogiannidis et al. (adaptive), and Reference (optimal). Each column has a corresponding histogram below it showing the distribution of samples. The images show various scenes, including a car interior, a person in a room, and a landscape with a bridge.

Fig. 1. We present a hierarchical path tracing framework in which a camera, light, and phone sensors to generate high-quality samples for rendering. The figure shows a comparison of different sampling strategies. The columns represent different methods: Path (baseline), Sobol (low-discrepancy), Molar (adaptive), Molt (adaptive), Ravi et al. (adaptive), Zhou et al. (adaptive), Paliogiannidis et al. (adaptive), and Reference (optimal). Each column has a corresponding histogram below it showing the distribution of samples. The images show various scenes, including a car interior, a person in a room, and a landscape with a bridge.

Fig. 1. We present a hierarchical path tracing framework in which a camera, light, and phone sensors to generate high-quality samples for rendering. The figure shows a comparison of different sampling strategies. The columns represent different methods: Path (baseline), Sobol (low-discrepancy), Molar (adaptive), Molt (adaptive), Ravi et al. (adaptive), Zhou et al. (adaptive), Paliogiannidis et al. (adaptive), and Reference (optimal). Each column has a corresponding histogram below it showing the distribution of samples. The images show various scenes, including a car interior, a person in a room, and a landscape with a bridge.

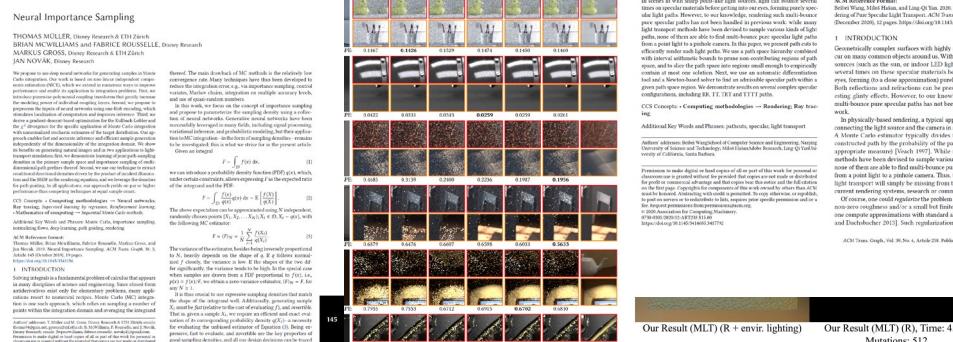
Path Cuts: Efficient Rendering of Pure Specular Light Transport

BEIBEI WANG, School of Computer Science and Engineering, Nanjing University of Science and Technology
MILOS HASAN, Adobe Research

LING-QI YAN, University of California, Santa Barbara

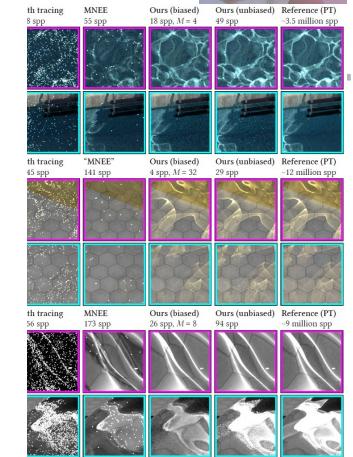
Path Graphs: Iterative Path Space Filtering

YI DENG, Cornell University, USA
AHMET HASAN, Adobe Research, USA
KATHARINA KREUZER, Cornell University, USA
ZEMING XU, Adobe Research, USA
STEVE MARSHNER, Cornell University, USA



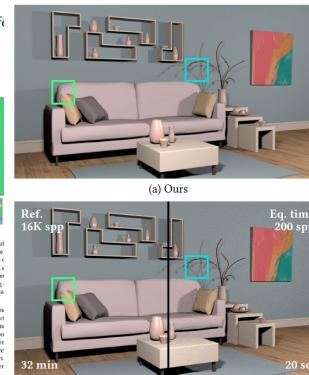
Specular Manifold Sampling for Rendering High-Frequency Caustics and Gl

TIZIAN ZELTNER, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland
ILIYAN GEORGIEV, Autodesk, United Kingdom
WENZEL JAKOB, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland



, prior-work MNEE [Hanika et al. 2015a], and both unbiased and biased versions of our one connection to the light source via the specular interface, whereas our techniques used way. The latter removes some of the high-frequency noise introduced by unbiased set size parameter M . We report samples per pixel (spp) computed by each method, as well as

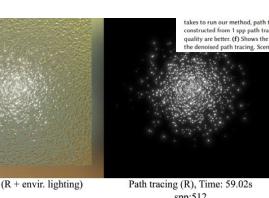
probability estimation and trades it for energy loss with the trial-set size parameter M . We report samples per pixel (spp) computed by each method, as well as the chosen M in the biased case.



Vectorization for Fast, Analytic, and Diff_I
YANG ZHOU, University of California, Santa Barbara
LIFAN WU, NVIDIA
RAVI RAMAMOORTHI, University of California, San Diego
UNO CHANG, University of California, Berkeley



Ours (44 sec) PT ref. (65K spp) PT eq. time (370 spp) FD ref. (256K)



One-Day Tutorial of Path Tracing



“Too hard and too many methods to study”

One-Day Tutorial of Path Tracing



“Too hard and too many methods to study”

- We can learn **path tracing** in one day
- Only use **Python** based on **Mitsuba 3**

One-Day Tutorial of Path Tracing



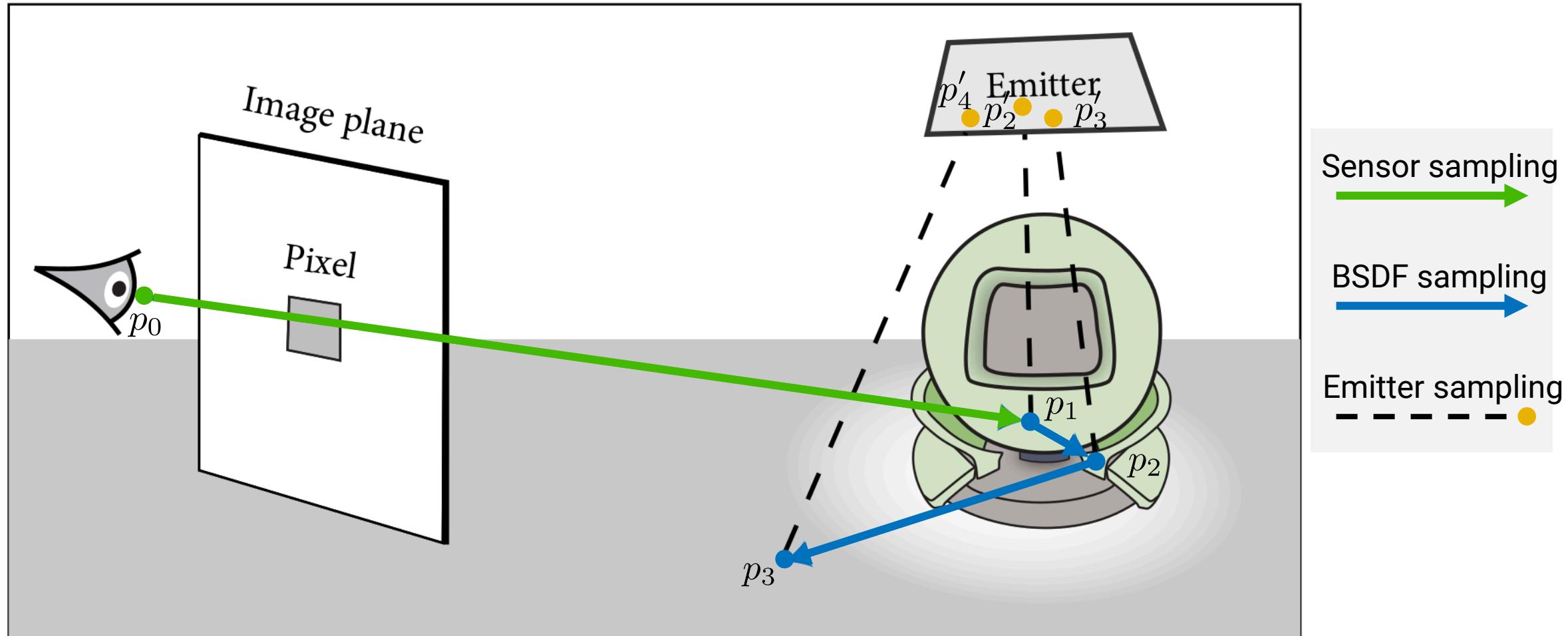
1. What should we compute?
 - Physics: radiometry

2. How can we compute?
 - Numerical method: Monte Carlo integration

One-Day Tutorial of Path Tracing

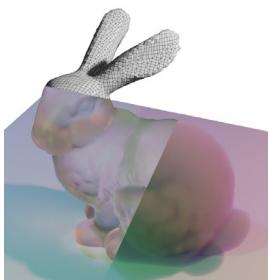


3. Path Tracing Algorithm



[https://mitsuba.readthedocs.io/en/latest/_images/integrator_path_figure.png]

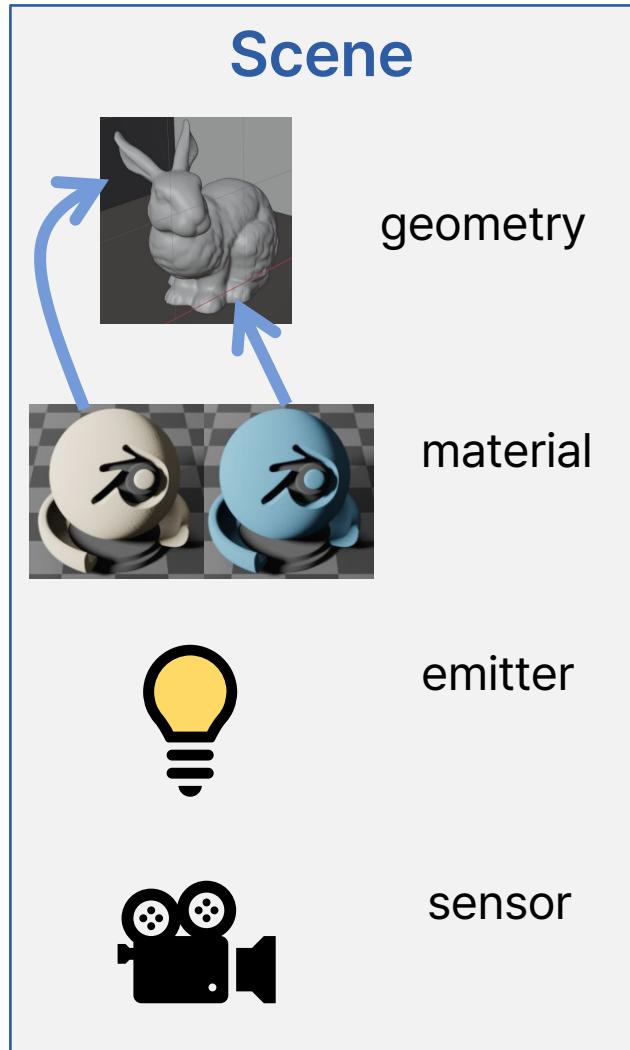
1. What should we compute?



→ Radiometry

- What is radiance?

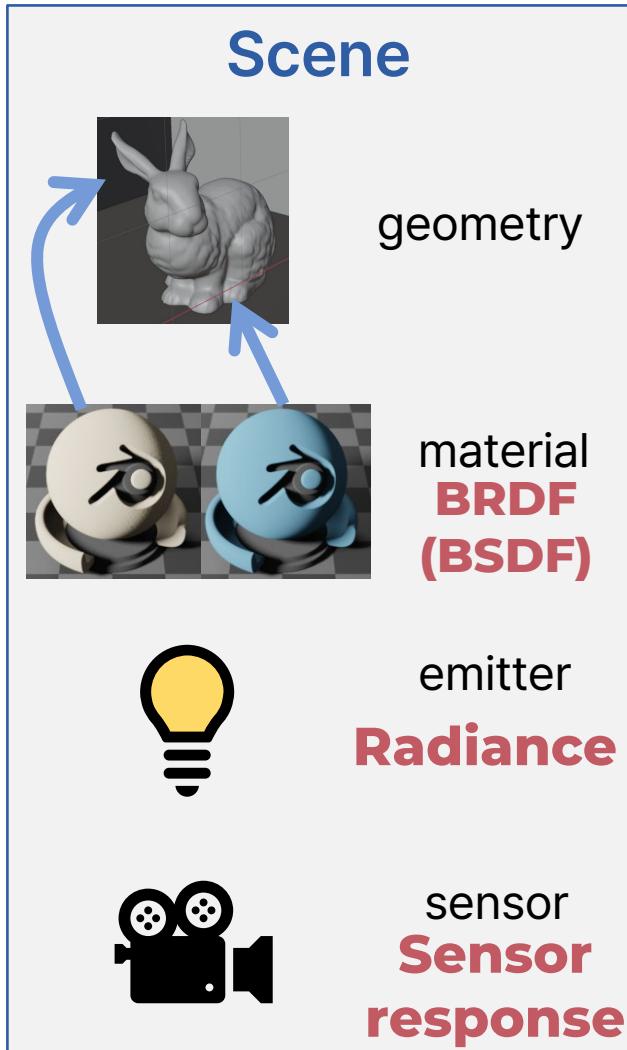
Rendering



Which physical quantities?

Which physical quantities?

Rendering



Which physical law?

Rendering equation
(+ constant radiance along ray)
(+ measurement equation)

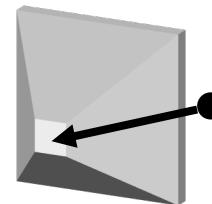
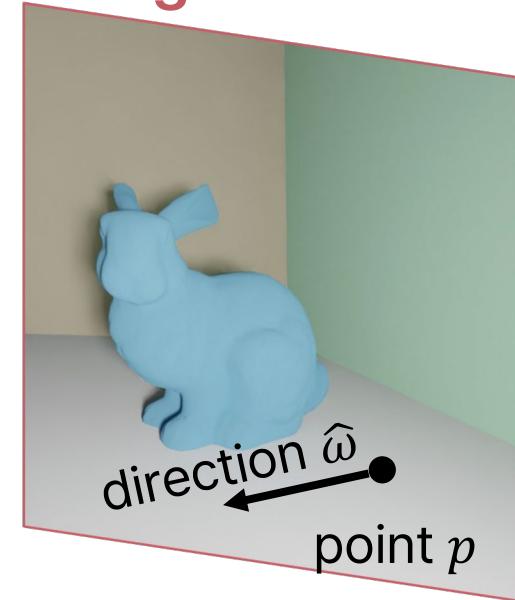


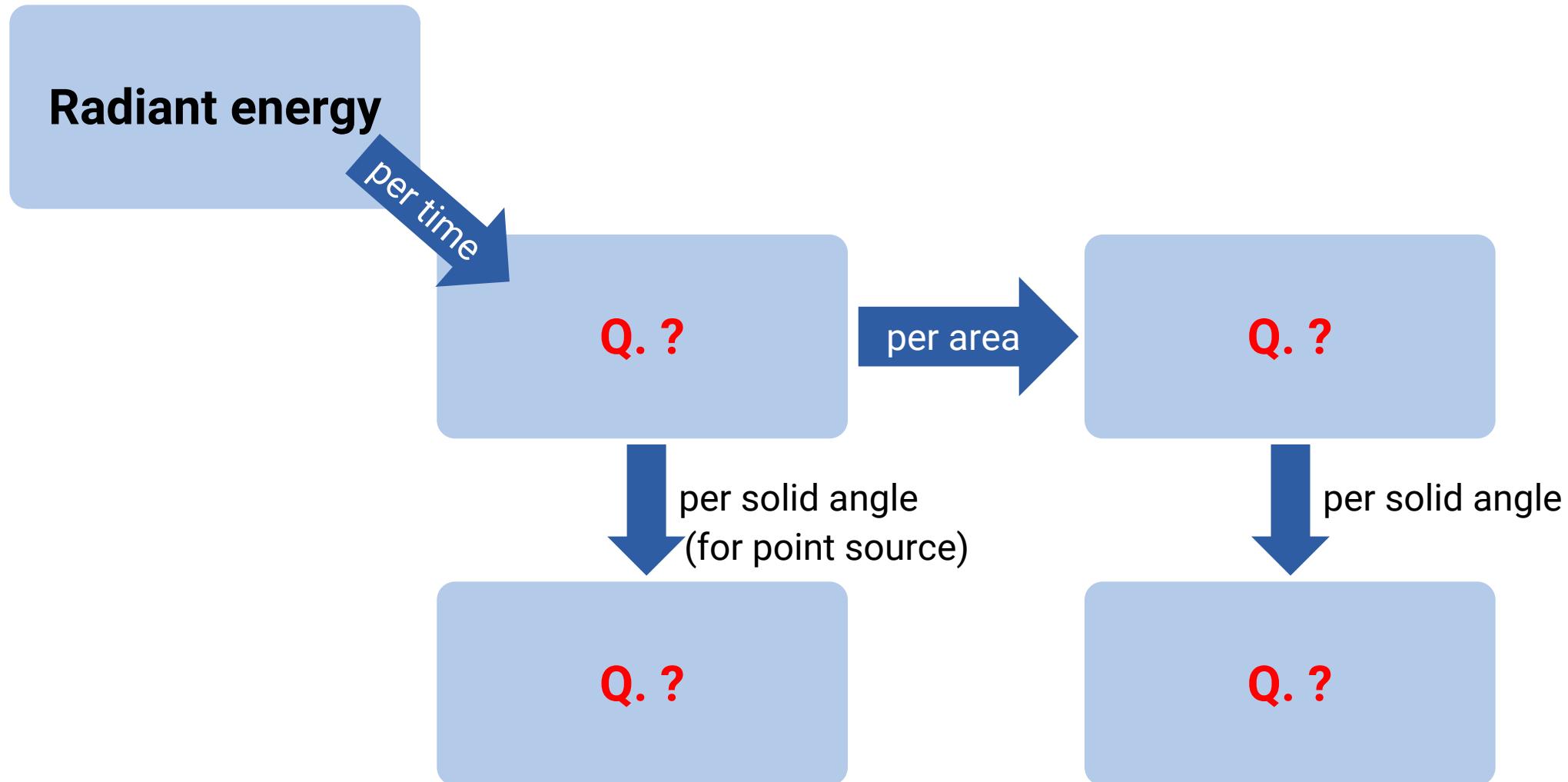
Image $\mathbf{I} \in \mathbb{R}^{H \times W}$



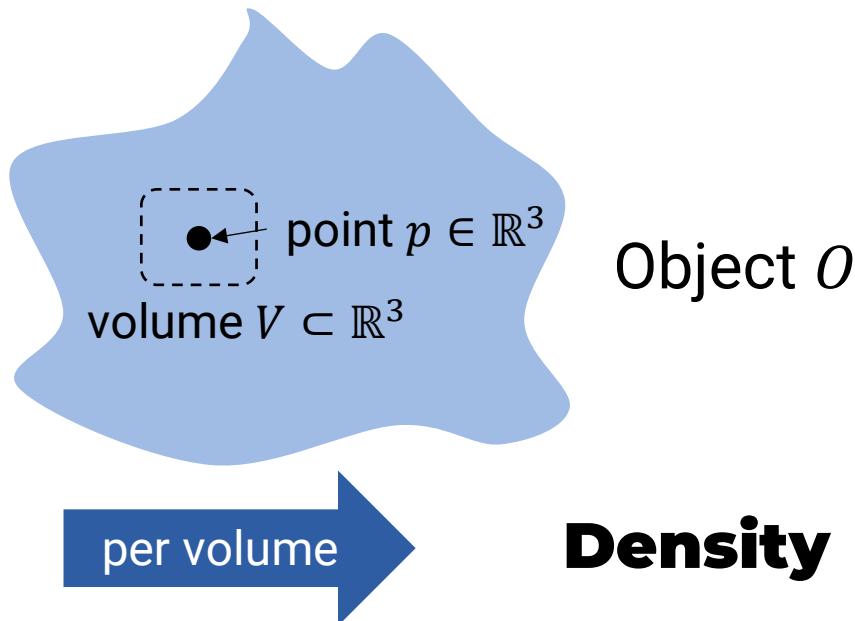
Which physical quantities?
Radiance at p along $\hat{\omega}$

Which physical quantities?

Radiometric quantities

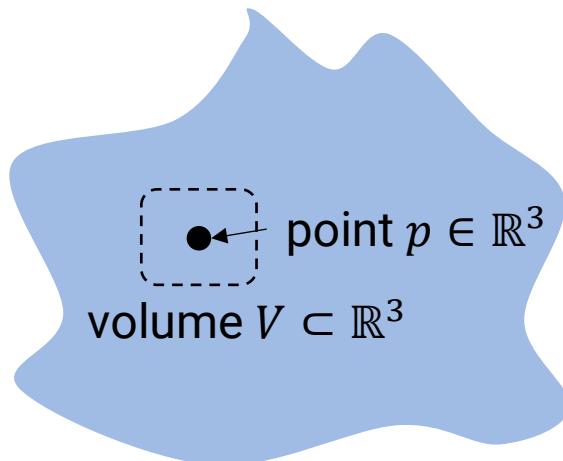


Concepts of mass vs. density



- | | |
|--|--|
| <p>✓ Mass of the object O</p> <p>✓ Mass of some region (volume) V</p> <p>✗ Mass at the point p
→ illegal or meaningless (always zero)</p> | <p>✗ Density of the object O
→ illegal or “average density” of the object O</p> <p>✗ Density of some region (volume) V
→ illegal or “average density” of the volume V</p> <p>✓ Density at the point p</p> |
|--|--|

Concepts of mass vs. density

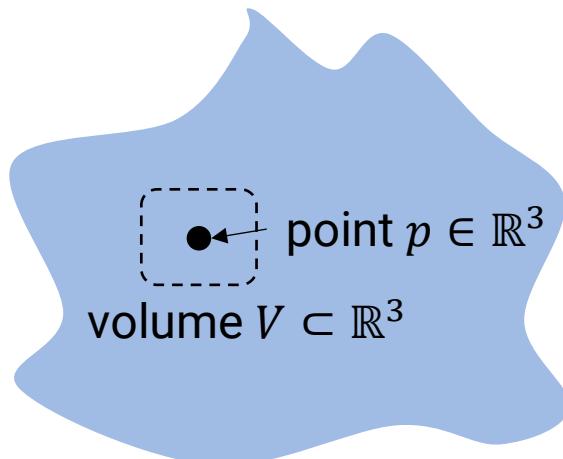


Mass of *what*?

- “mass of an object O ”
= “mass of the volume of O ”

Density of *what*?

Concepts of mass vs. density



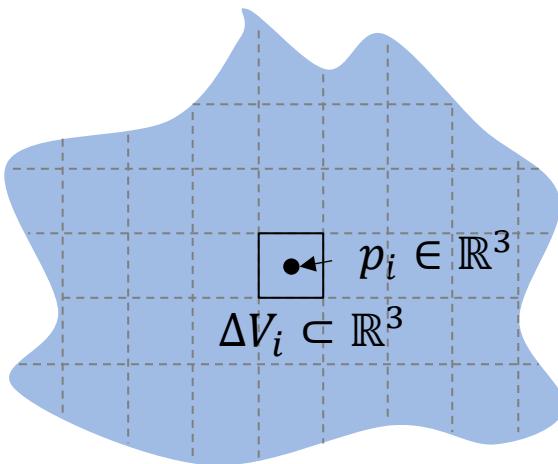
Mass of $V \subset \mathbb{R}^3$

Density of $p \in \mathbb{R}^3$

$\text{mass}(V)$

per volume → $\text{density}(p) = \lim_{\substack{\text{vol}(V) \rightarrow 0 \\ p \in V}} \frac{\text{mass}(V \text{ [dashed box]})}{\text{vol}(V \text{ [dashed box]})}$

Concepts of mass vs. density



Mass of $V \subset \mathbb{R}^3$

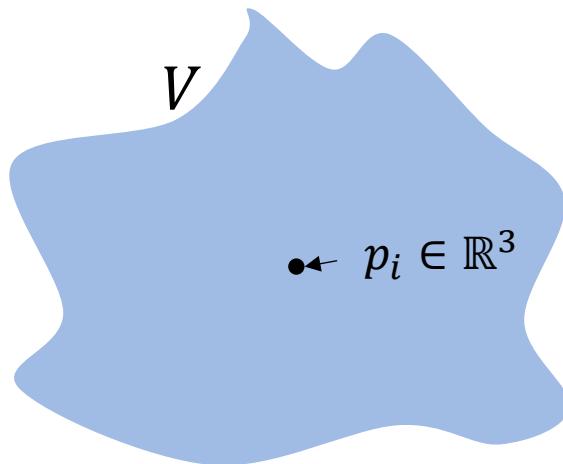
$$\begin{aligned}\text{mass}(V) &= \sum_i \text{mass}(\Delta V_i) \\ &\approx \sum_i \underset{\substack{\rightarrow \\ \text{vol}(\Delta V_i) \rightarrow 0}}{\text{density}(p_i)} \text{vol}(\Delta V_i)\end{aligned}$$

← over volume

Density of $p \in \mathbb{R}^3$

$$\text{density}(p) = \lim_{\substack{\text{vol}(V) \rightarrow 0 \\ p \in V}} \frac{\text{mass}(V)}{\text{vol}(V)}$$

Concepts of mass vs. density



Mass of $V \subset \mathbb{R}^3$ [kg]

Density at $p \in \mathbb{R}^3$ [kg/m³]

Proposition

Relations between mass m of a volume $V \subset \mathbb{R}^3$ and density ρ at a point $p \in \mathbb{R}^3$ is that:

$$m(V) = \int_V \rho(p) dp$$

per volume →
← over volume

$$\rho(p) = \lim_{\substack{|V| \rightarrow 0 \\ p \in V}} \frac{m(V)}{|V|}$$

Concepts of mass vs. density



Notation comparison

Other text often write $\frac{dm}{dV}$ instead of $\lim_{|V| \rightarrow 0, p \in V} \frac{m(V)}{|V|}$ but the former notation may give a misunderstanding that m is a function of a real number (volume measure) rather than one of a subset of \mathbb{R}^3 (volume region). The formula $\frac{dm}{dV}$ can be correctly understood only if it denoted a Radon-Nikodym derivative, which is dealt in *measure theory* (4th grade in Math. major).

We do not assume measure theory as a prerequisite, so we use the latter notation $\lim_{|V| \rightarrow 0, p \in V} \frac{m(V)}{|V|}$ for explicitness.

Proposition

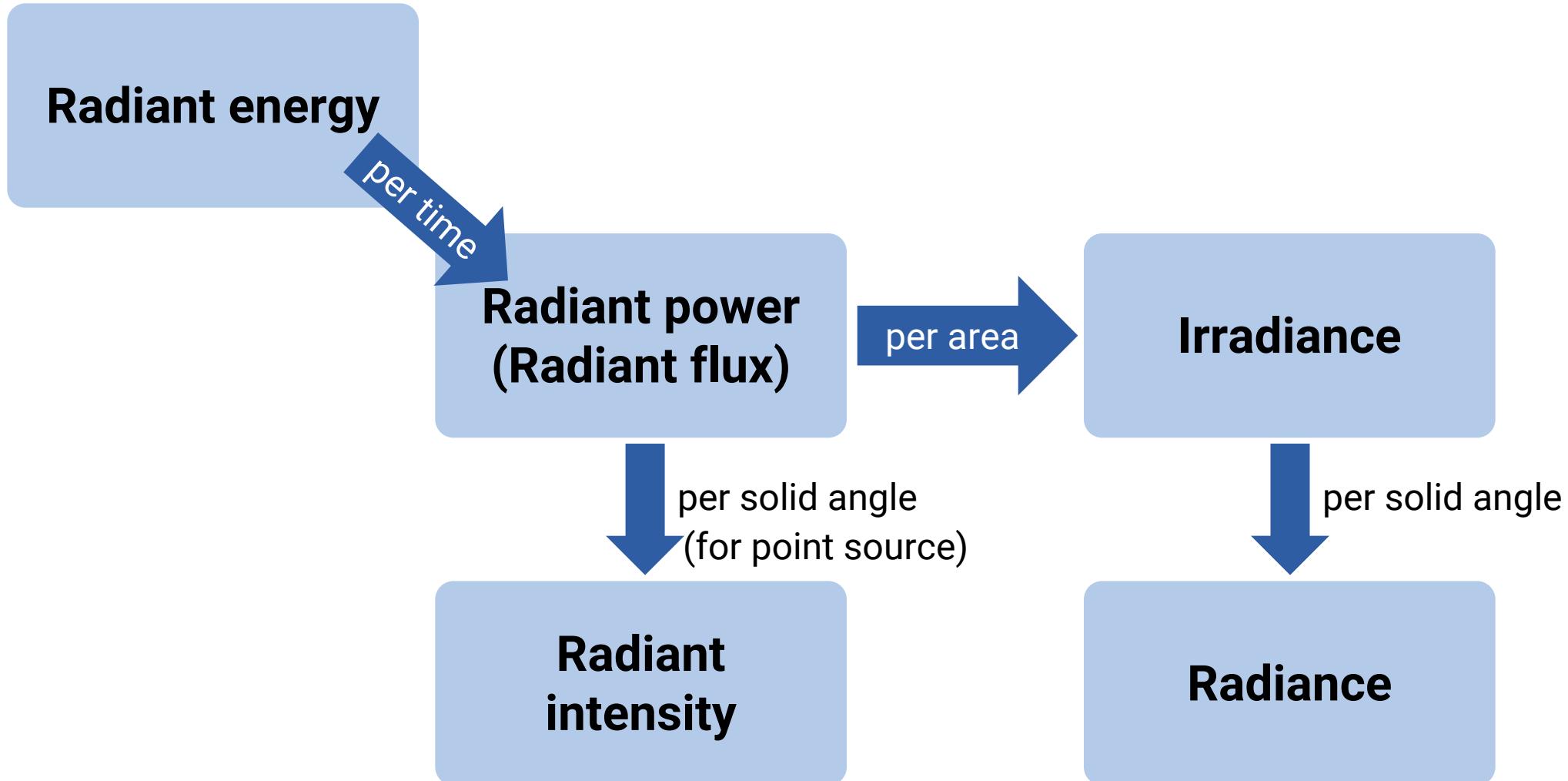
Relations between mass m of a volume $V \subset \mathbb{R}^3$ and density ρ at a point $p \in \mathbb{R}^3$ is that:

$$m(V) = \int_V \rho(p) dp$$

per volume →
← over volume

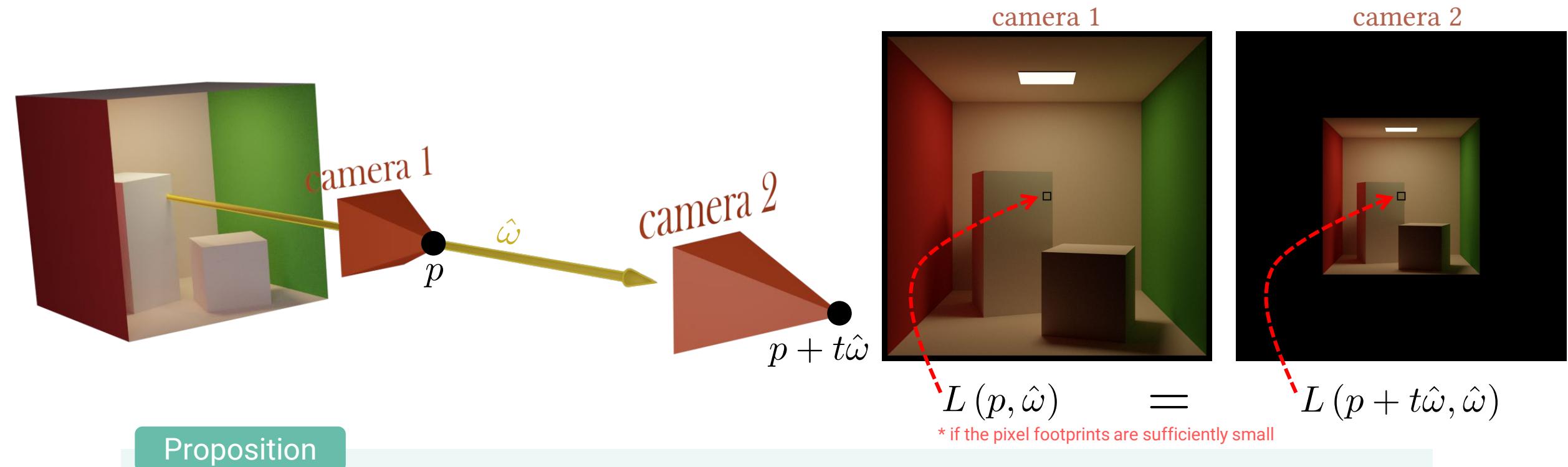
$$\rho(p) = \lim_{\substack{|V| \rightarrow 0 \\ p \in V}} \frac{m(V)}{|V|}$$

Radiometric quantities



See the full slides for more details:

- <https://github.com/shinyoung-yi/lecture-rendering-mitsuba/blob/main/slides/01.%20radiometry%20and%20light%20transport.pdf>



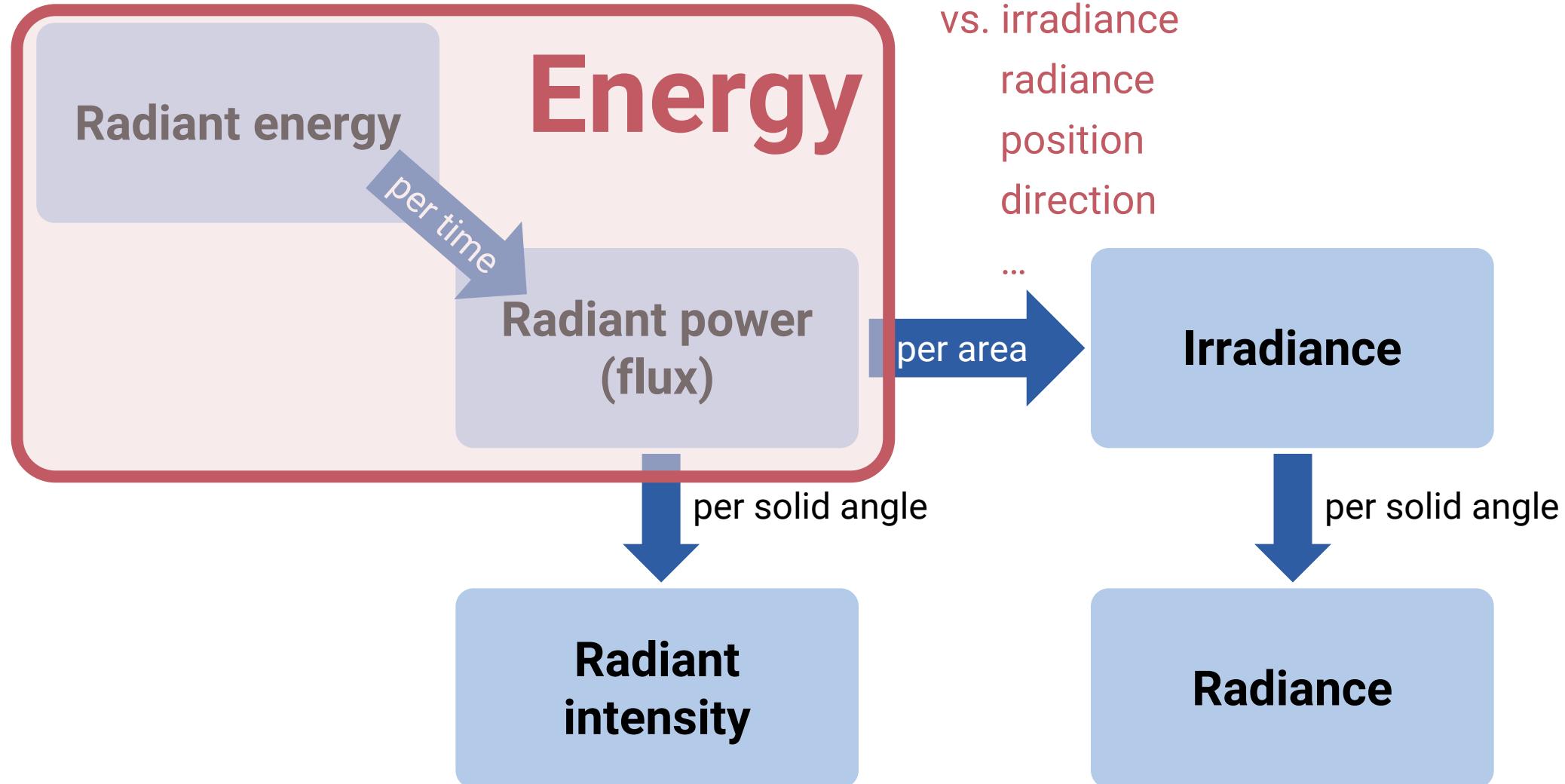
Proposition

Radiance is invariant along ray:

$$L(p, \hat{\omega}) = L(p + t\hat{\omega}, \hat{\omega}) \quad \forall t \in \mathbb{R}$$

whenever there is no material between p and $p + t\hat{\omega}$

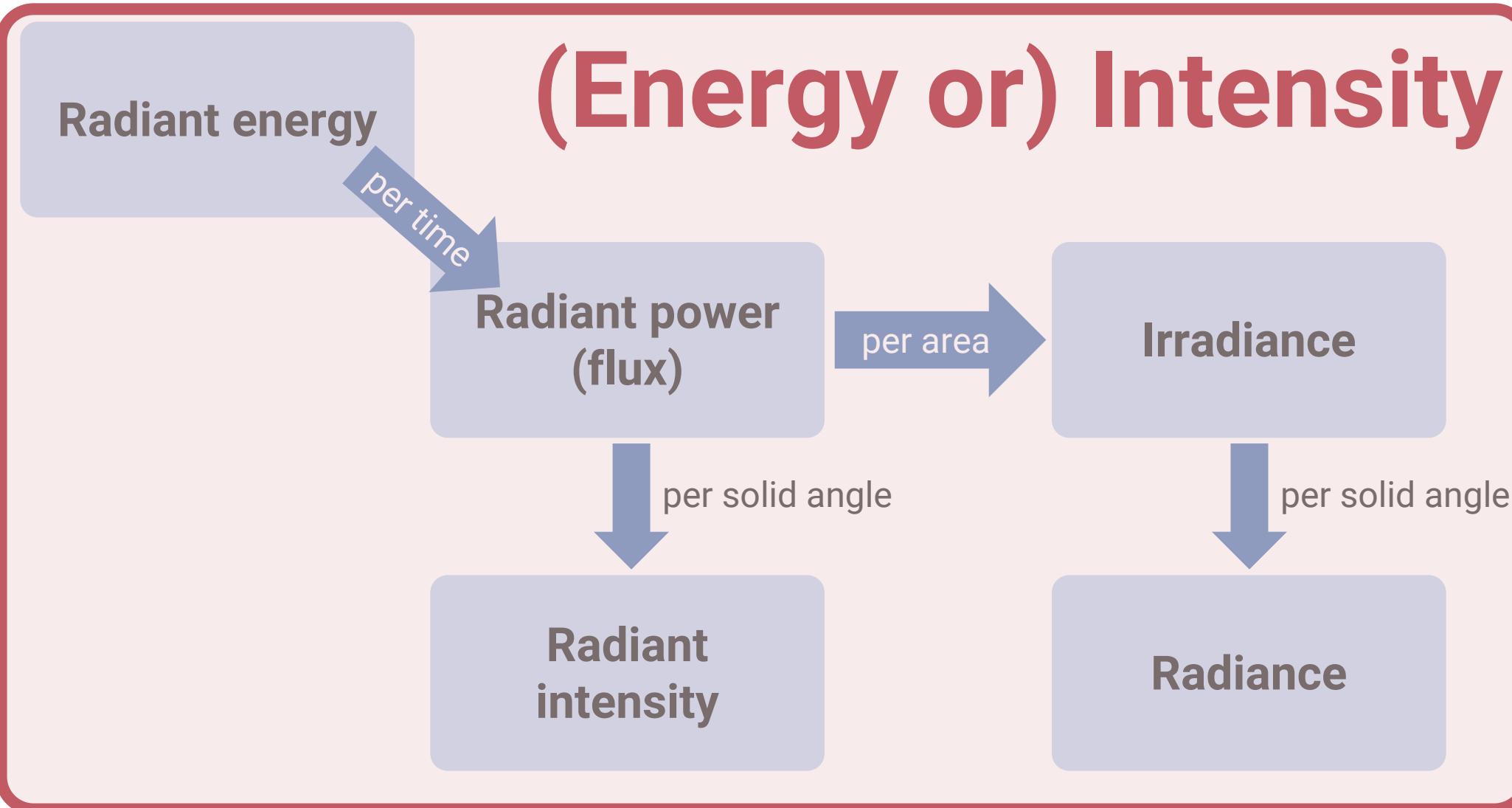
Slight abuse of terminology



Slight abuse of terminology



vs. position
direction
...





Radiant energy

per time

Radiant power
(flux)

per area

Irradiance

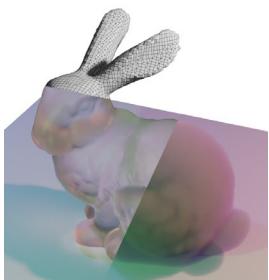
per solid angle

**Radiant
intensity**

per solid angle

Radiance

1. What should we compute?



→ Radiometry

- What is radiance?
- BRDFs (BSDFs) and rendering equation

Material Appearance



How can we characterize material appearance (reflectance)?

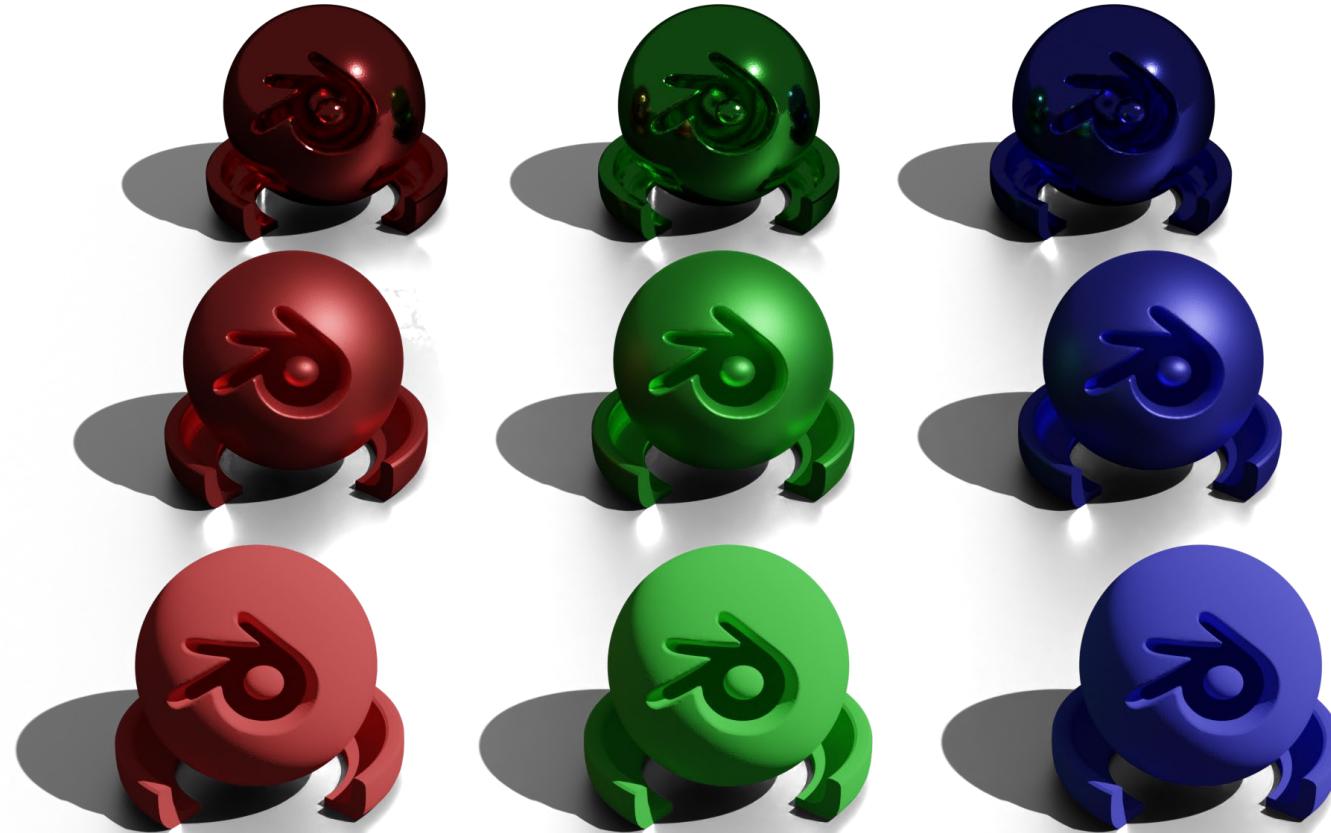


Is single numbers of reflectance per RGB channel (or wavelength) enough?

Material Appearance

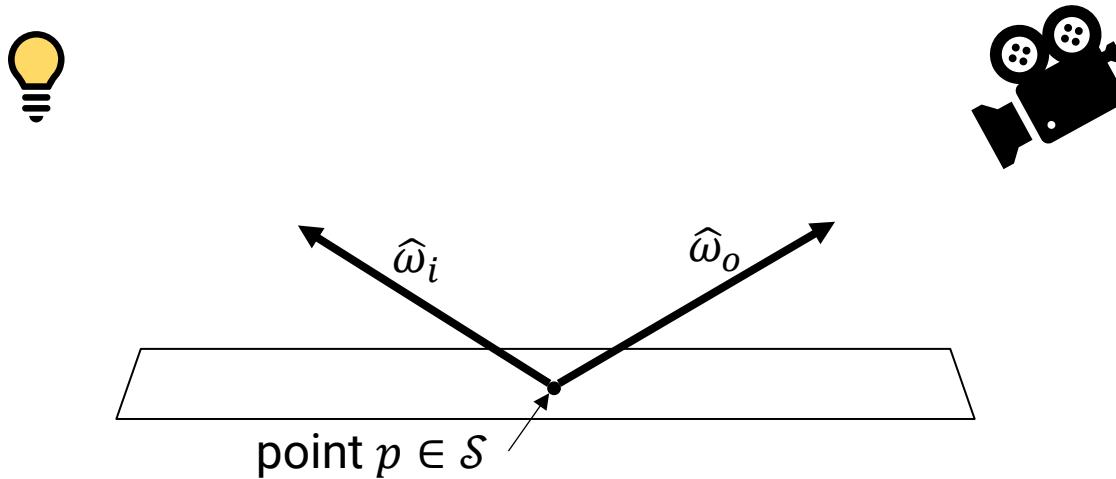


How can we characterize material appearance (reflectance)?



Is single numbers of reflectance per RGB channel (or wavelength) enough?

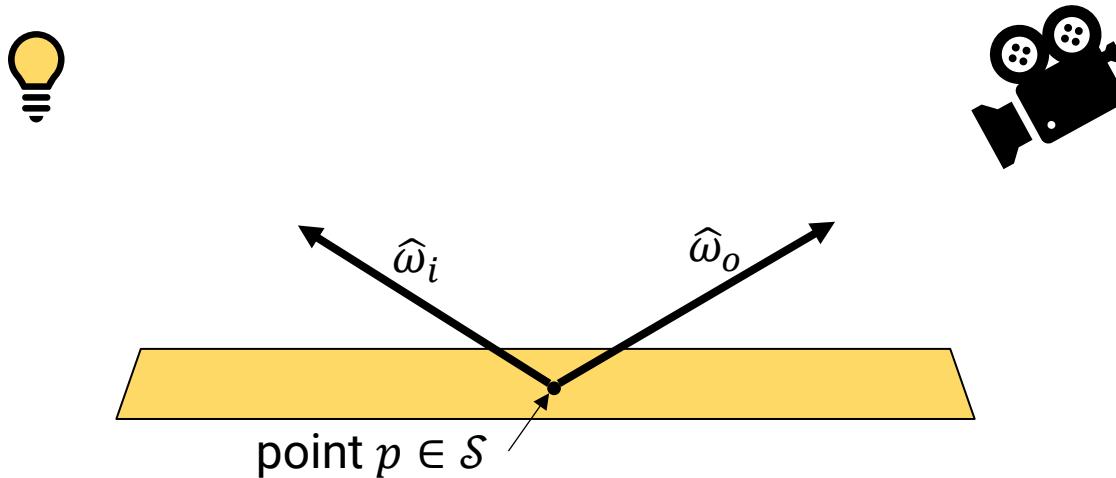
Rendering equation



From definition of BRDFs...

$$L^{(\text{out})}(p, \hat{\omega}_o) = \int_{\mathbb{S}^2} L^{(\text{in})}(p, \hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation



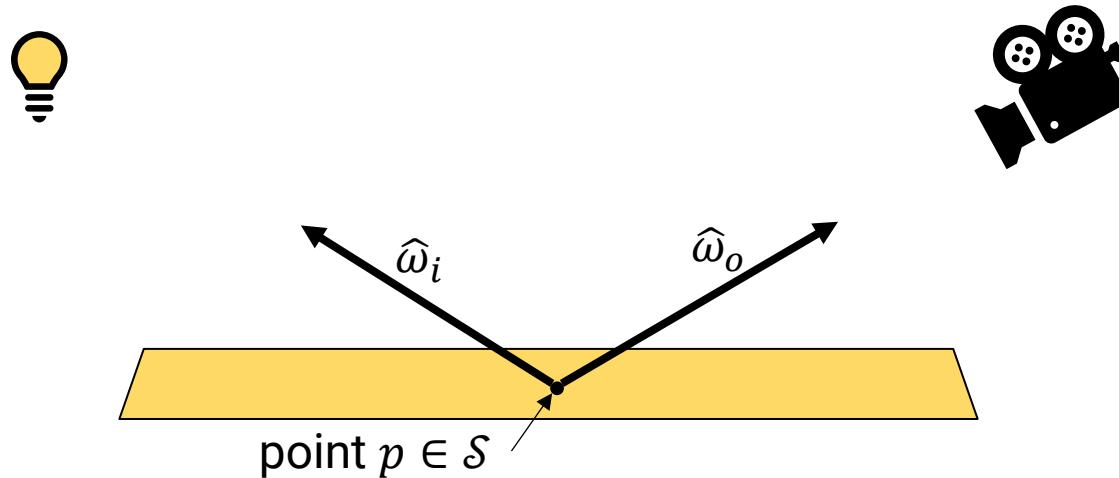
From definition of BRDFs...

+ Emission

Rendering equation
(light transport equation)

$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \hat{\omega}_o) + \int_{\mathbb{S}^2} L^{(\text{in})}(p, \hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation

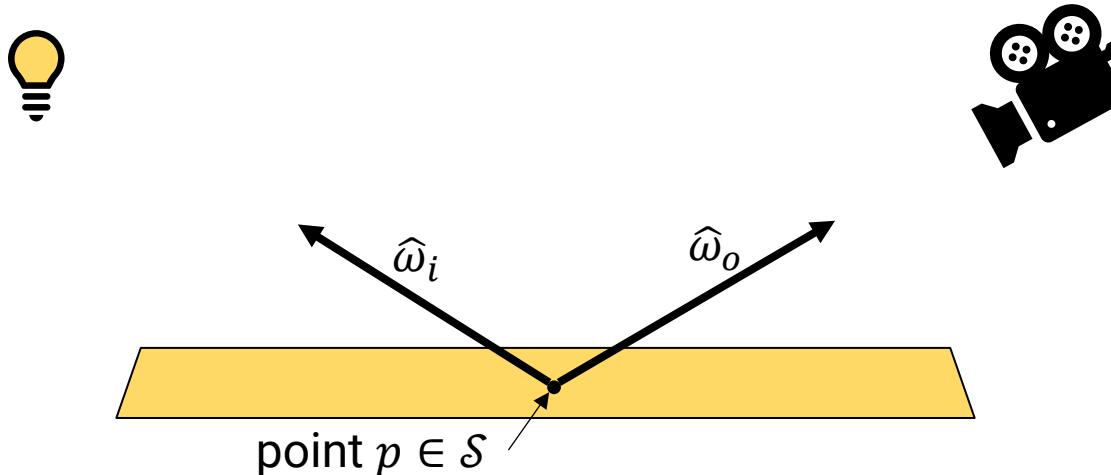


Q. What are knowns and unknowns?

Rendering equation
(light transport equation)

$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \hat{\omega}_o) + \int_{\mathbb{S}^2} L^{(\text{in})}(p, \hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation



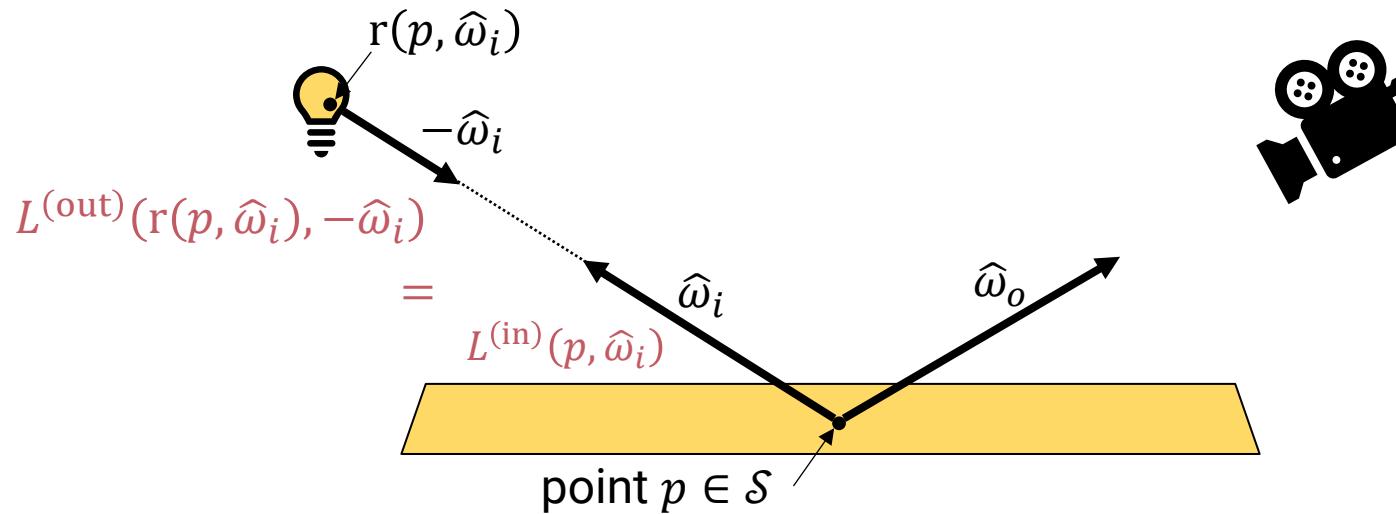
Q. What are knowns and unknowns?

Rendering equation
(light transport equation)

$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \hat{\omega}_o) + \int_{\mathbb{S}^2} L^{(\text{in})}(p, \hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

The term $L^{(\text{out})}(p, \hat{\omega}_o)$ is labeled "unknown". The terms $L_e(p, \hat{\omega}_o)$ and $\int_{\mathbb{S}^2} L^{(\text{in})}(p, \hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$ are labeled "known".

Rendering equation

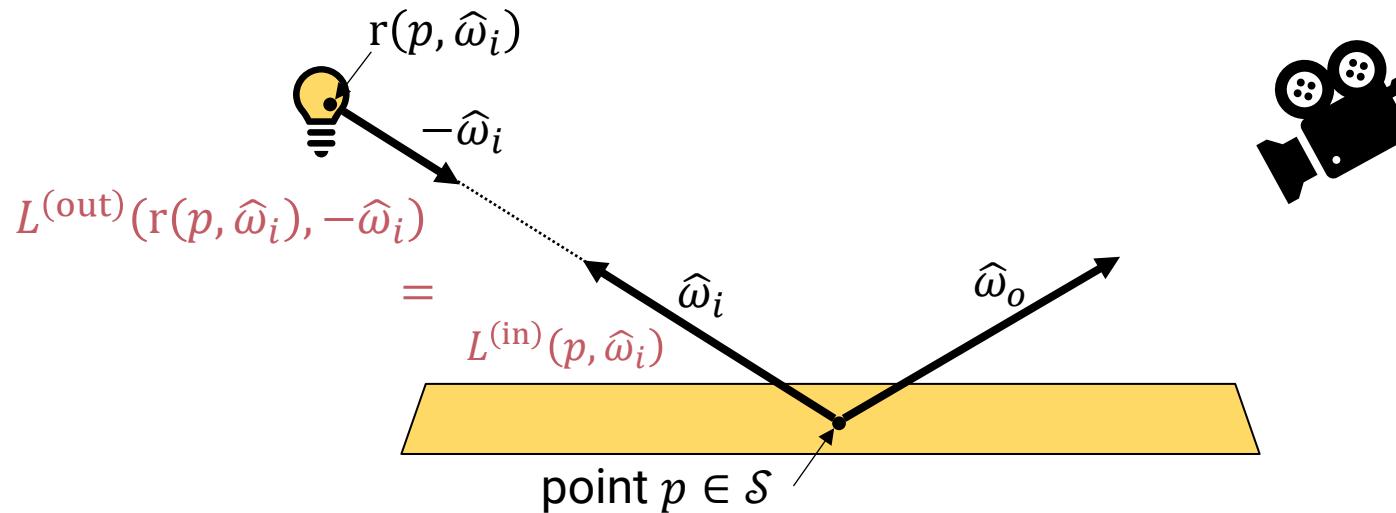


Q. Any relationship between $L^{(\text{out})}$ and $L^{(\text{in})}$?

Rendering equation
(light transport equation)

$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \hat{\omega}_o) + \int_{\mathbb{S}^2} L^{(\text{in})}(p, \hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation

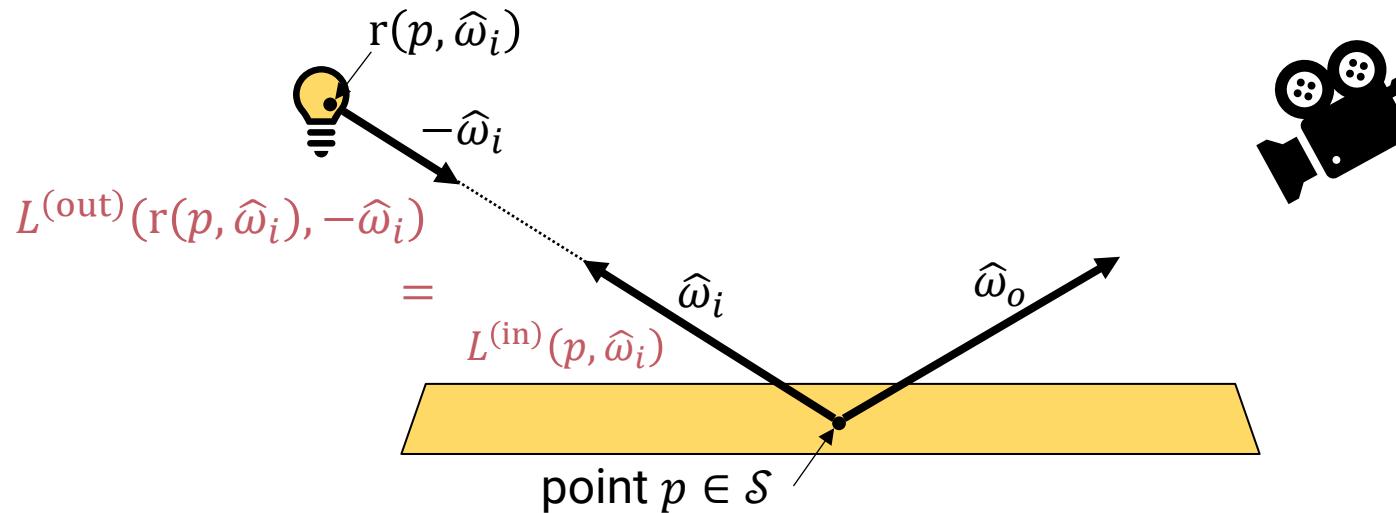


Q. Any relationship between $L^{(out)}$ and $L^{(in)}$?

Rendering equation
(light transport equation)

$$L^{(out)}(p, \hat{\omega}_o) = L_e(p, \hat{\omega}_o) + \int_{\mathbb{S}^2} L^{(out)}(r(p, \hat{\omega}_i), -\hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation



Q. Any relationship between $L^{(out)}$ and $L^{(in)}$?

Rendering equation
(light transport equation)

$$L^{(out)}(p, \hat{\omega}_o) = L_e(p, \hat{\omega}_o) + \int_{\mathbb{S}^2} L^{(out)}(r(p, \hat{\omega}_i), -\hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation



$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \omega_{\hat{o}}) + \int_{\mathbb{S}^2} L^{(\text{out})}(r(p, \hat{\omega}_i), -\hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Recursive!

$$L(p, \hat{\omega}_o) = L_e + \int_{\mathbb{S}^2} L f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation



$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \omega_{\hat{o}}) + \int_{\mathbb{S}^2} L^{(\text{out})}(r(p, \hat{\omega}_i), -\hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Recursive!

$$L(p, \hat{\omega}_o) = L_e + \int_{\mathbb{S}^2} \left[L'_e + \int_{\mathbb{S}^2} L f'_s |\hat{n} \cdot \hat{\omega}'_i| d\hat{\omega}'_i \right] f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Rendering equation



$$L^{(\text{out})}(p, \hat{\omega}_o) = L_e(p, \omega_{\hat{o}}) + \int_{\mathbb{S}^2} L^{(\text{out})}(r(p, \hat{\omega}_i), -\hat{\omega}_i) f_s(p, \hat{\omega}_i, \hat{\omega}_o) |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

Recursive!

$$L(p, \hat{\omega}_o) = L_e + \int_{\mathbb{S}^2} \left[L'_e + \int_{\mathbb{S}^2} \left[L''_e + \int_{\mathbb{S}^2} L f_s'' |\hat{n} \cdot \hat{\omega}_i''| d\hat{\omega}_i'' \right] f'_s |\hat{n} \cdot \hat{\omega}_i'| d\hat{\omega}_i' \right] f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i$$

...

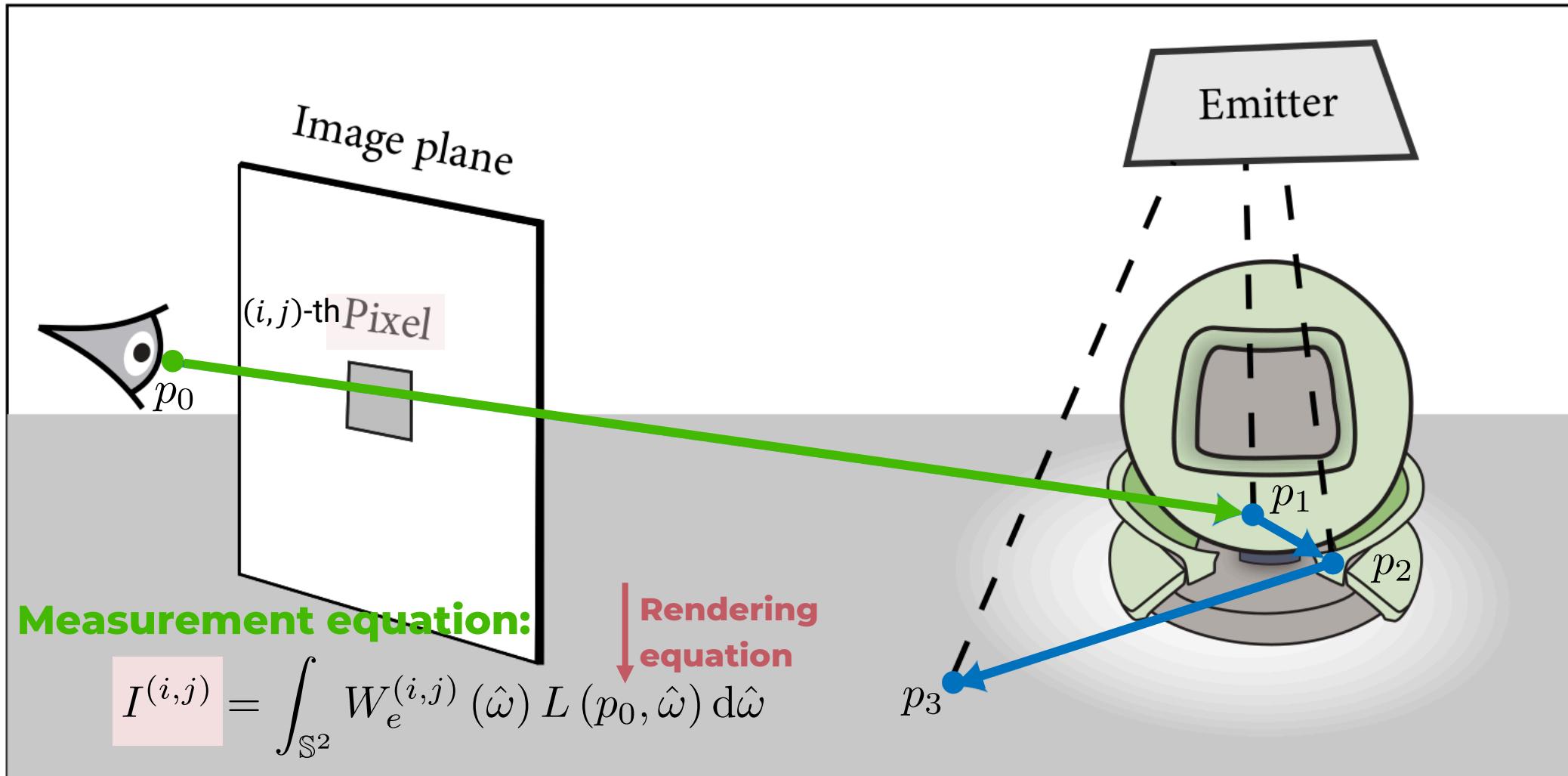
$$\begin{aligned} L(p, \hat{\omega}_o) &= L_e \\ &\quad + \int_{\mathbb{S}^2} L'_e f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i \\ &\quad + \int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} L''_e f'_s |\hat{n} \cdot \hat{\omega}_i'| d\hat{\omega}_i' \right] f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i \\ &\quad + \int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} L'''_e f''_s |\hat{n} \cdot \hat{\omega}_i''| d\hat{\omega}_i'' \right] f'_s |\hat{n} \cdot \hat{\omega}_i'| d\hat{\omega}_i' \right] f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i \end{aligned}$$

Rendering equation



$$\begin{aligned} L(p, \hat{\omega}_o) = & L_e \\ & + \int_{\mathbb{S}^2} L'_e f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i \\ & + \int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} L''_e f'_s |\hat{n} \cdot \hat{\omega}'_i| d\hat{\omega}'_i \right] f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i \\ & + \int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} L'''_e f''_s |\hat{n} \cdot \hat{\omega}''_i| d\hat{\omega}''_i \right] f'_s |\hat{n} \cdot \hat{\omega}'_i| d\hat{\omega}'_i \right] f_s |\hat{n} \cdot \hat{\omega}_i| d\hat{\omega}_i \end{aligned}$$

Notation & Measurement Equation



[https://mitsuba.readthedocs.io/en/latest/_images/integrator_path_figure.png]



Rendering equation

$$\begin{aligned} L(p_1, \hat{\omega}_{10}) &= L_{e,10} \\ &+ \int_{\mathbb{S}^2} L_{e,21} f_{s,210}^\perp d\hat{\omega}_{12} \\ &+ \int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} L_{e,32} f_{s,321}^\perp d\hat{\omega}'_i \right] f_{s,210}^\perp d\hat{\omega}_{23} \\ &+ \int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} \left[\int_{\mathbb{S}^2} L_{e,43} f_{s,432}^\perp d\hat{\omega}''_i \right] f_{s,321}^\perp d\hat{\omega}'_i \right] f_{s,210}^\perp d\hat{\omega}_{34} \end{aligned}$$

Rendering equation



$$\begin{aligned} L(p_1, \hat{\omega}_{10}) &= L_{e,10} \\ &+ \int_{\mathbb{S}^2} f_{s,210}^\perp L_{e,21} d\hat{\omega}_{12} \\ &+ \int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp L_{e,32} d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \\ &+ \int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \end{aligned}$$



Rendering equation

$$\begin{aligned} L(p_1, \hat{\omega}_{10}) = & L_{e,10} \\ & + \int_{\mathbb{S}^2} f_{s,210}^\perp L_{e,21} d\hat{\omega}_{12} \\ & + \int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp L_{e,32} d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \\ & + \int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \end{aligned}$$

Rendering & Measurement equation



$$\begin{aligned} I &= \int_{\mathbb{S}^2} W_e(\hat{\omega}_{10}) L(p_1, \hat{\omega}_{10}) d\hat{\omega}_{01} \\ &= \int_{\mathbb{S}^2} W_e(\hat{\omega}_{10}) L_{e,10} d\hat{\omega}_{01} \\ &\quad + \int_{\mathbb{S}^2} W_e(\hat{\omega}_{10}) \left[\int_{\mathbb{S}^2} f_{s,210}^\perp L_{e,21} d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \\ &\quad + \int_{\mathbb{S}^2} W_e(\hat{\omega}_{10}) \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp L_{e,32} d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \\ &\quad + \int_{\mathbb{S}^2} W_e(\hat{\omega}_{10}) \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \end{aligned}$$

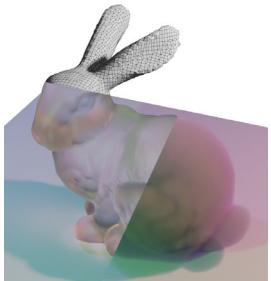
Integration domain: all paths (tuples of vertices)

integrand: path contribution $f(p_0 \dots p_j)$

2. How can we compute?

→ Monte Carlo Integration

- MC Integration and importance sampling

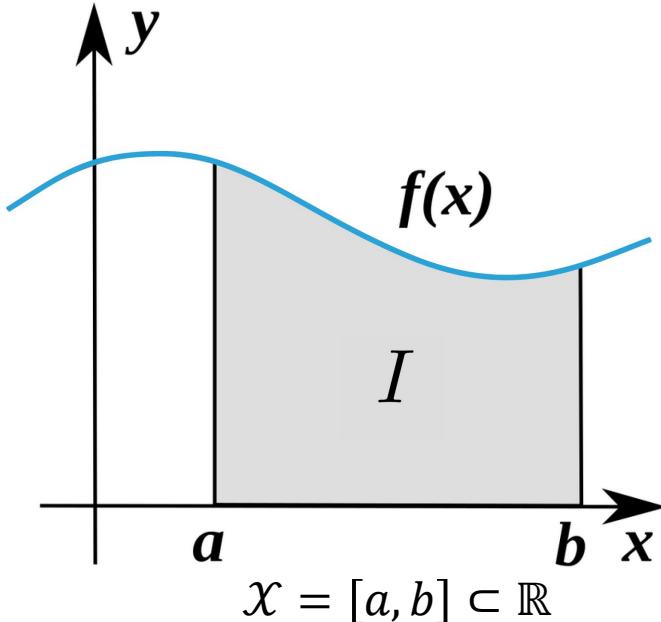


Integration



We have a function f on a domain \mathcal{X} .

Q. How can we compute $I = \int_{\mathcal{X}} f(x) dx$?



source: User 4C / Wikipedia
CC BY-SA 3.0

[Adam Celarek]

1. Analytic (symbolic) integration

- We can integrate only few functions analytically.
- Hardly predict difficulty of given problem

Try analytic integration for:

$$\int e^{-x^2} dx \quad \text{No}$$

$$\int xe^{-x^2} dx \quad \text{Yes}$$

$$\int x^2 e^{-x^2} dx \quad \text{No}$$

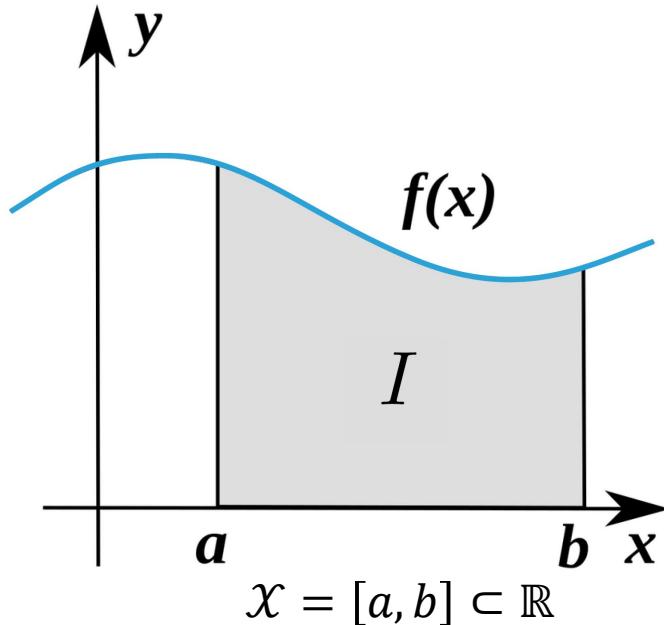
$$\int x^3 e^{-x^2} dx \quad \text{Yes}$$

Integration



We have a function f on a domain \mathcal{X} .

Q. How can we compute $I = \int_{\mathcal{X}} f(x) dx$?



source: User 4C / Wikipedia
CC BY-SA 3.0

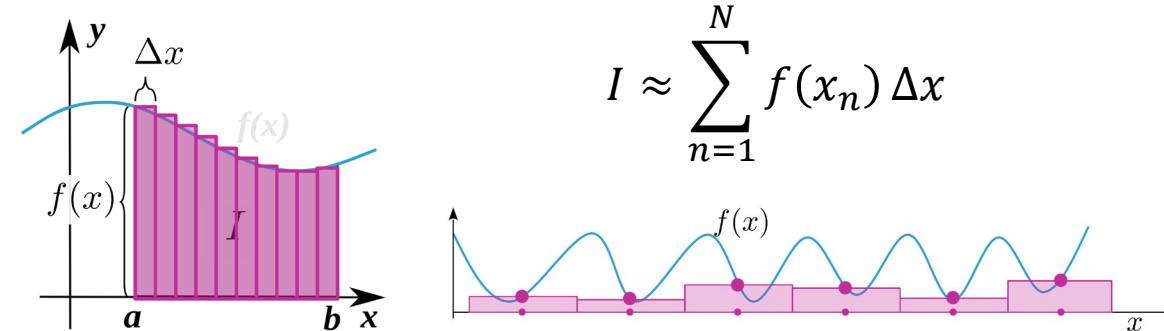
[Adam Celarek]

1. Analytic (symbolic) integration

- We can integrate only few functions analytically.
- Hardly predict difficulty of given problem

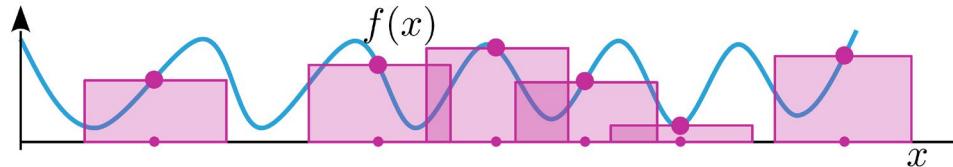
2. Numerical approximation by uniform grids

- For a fixed error level, we need $N^{\dim(\mathcal{X})}$ samples!
- Aliasing!



3. Monte Carlo estimation: for a random variable $X \sim \mathcal{U}(\mathcal{X})$

$$I \approx \hat{I} = \frac{|\mathcal{X}|}{N} \sum_{n=1}^N f(X_n)$$

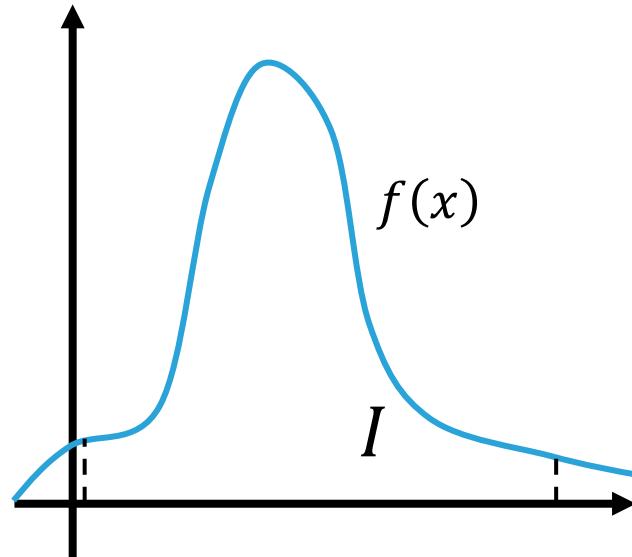


Monte Carlo Integration



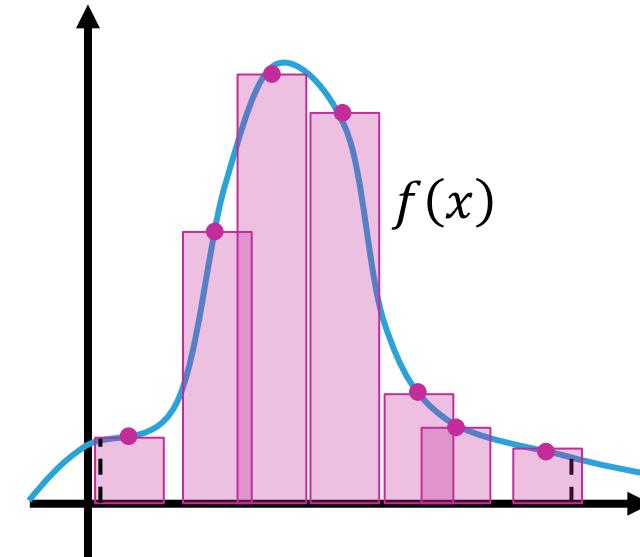
We have a function f on a domain \mathcal{X} .

Q. How can we compute $I = \int_{\mathcal{X}} f(x) dx$?



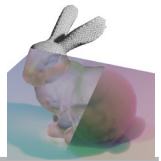
for a random variable $X \sim \mathcal{U}(\mathcal{X})$

$$I \approx \hat{I} = \frac{|\mathcal{X}|}{N} \sum_{n=1}^N f(X_n)$$



$$\text{Var}(\hat{I}) \propto \text{Var}(\square)$$

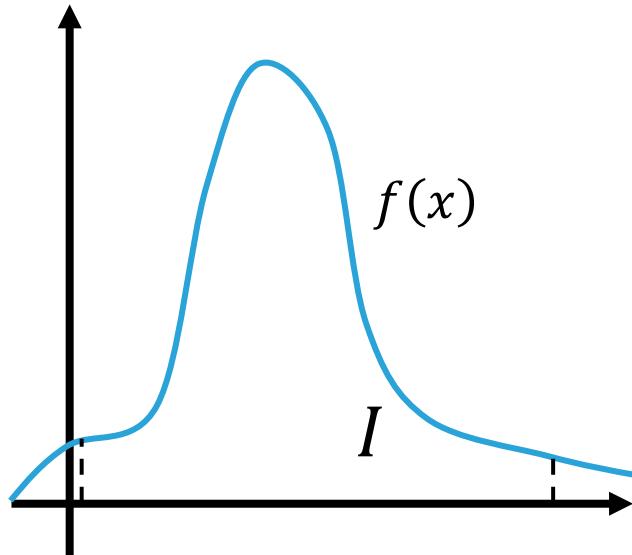
Monte Carlo Integration: importance sampling



for a random variable X with PDF hopefully $\propto f$

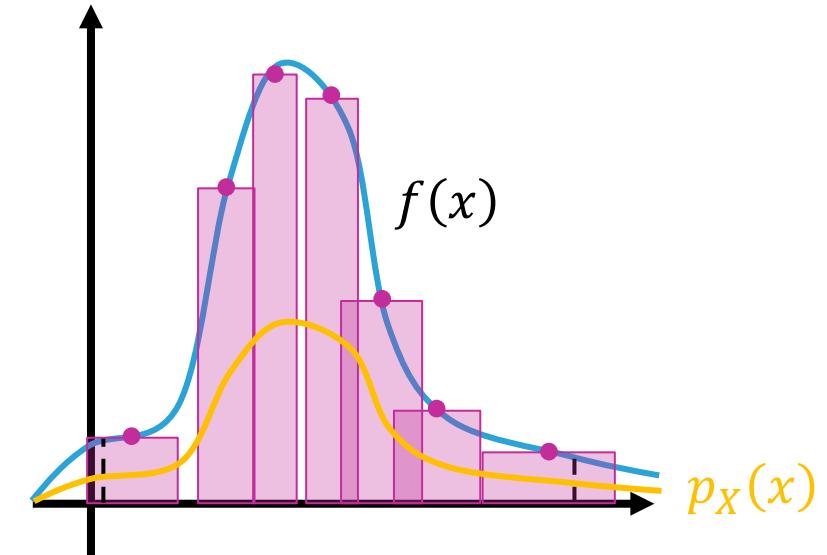
We have a function f on a domain \mathcal{X} .

Q. How can we compute $I = \int_{\mathcal{X}} f(x) dx$?



$$\mathcal{X} = [a, b] \subset \mathbb{R}$$

$$I \approx \hat{I} = \frac{1}{N} \sum_{n=1}^N \frac{f(X_n)}{p_X(X_n)}$$



$$Var(\hat{I}) \propto Var(\square)$$



Monte Carlo Integration



When is the MC integration correct (unbiased)?

$$\int_{\mathcal{X}} f(x) dx = I \approx \hat{I} = \frac{1}{N} \sum_{n=1}^N \frac{f(X_n)}{p_X(X_n)}$$

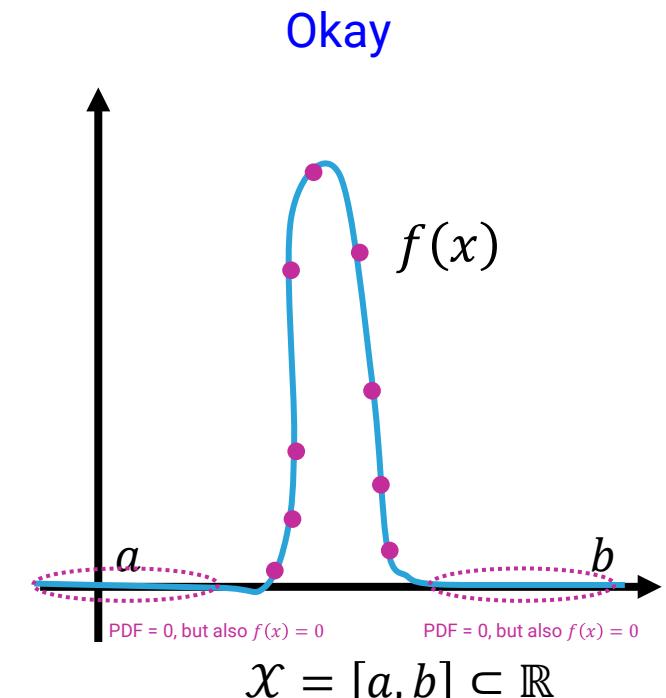
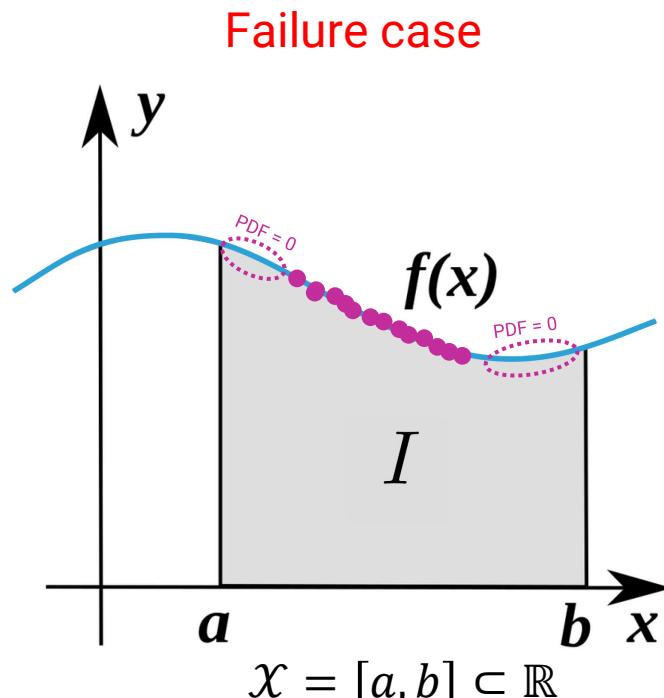
Monte Carlo Integration



When is the MC integration correct (unbiased)?

$$\int_{\mathcal{X}} f(x) dx = I = \mathbb{E}[\hat{I}] = \mathbb{E}\left[\frac{1}{N} \sum_{n=1}^N \frac{f(X_n)}{p_X(X_n)} \right]$$

For any $x \in \mathcal{X}$, if $f(x) \neq 0$ then $p_X(x) > 0$



Language of Monte Carlo Integration



Sampling: generate a realization of a random variable

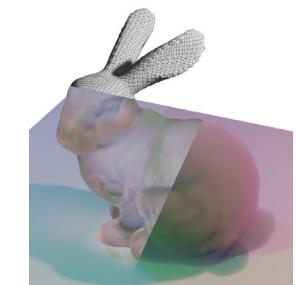
Which PDF do we choose?

How draw a sample of a r.v. with such PDF?

- “*This sampling strategy is good for given integrand f .*”
- “*In path tracing, BSDF sampling and emitter samplings are used.*”

Correct → **unbiased** (or **consistent**, at least)

Efficient → **low variance**
(less noise)



3. Path Tracing



Rendering

Rasterization

- OpenGL / DirectX
- Many techniques learned in undergrad-graphics courses

**Precomputed
radiance transfer**

Radiosity

Vectorization

[Zhou et al. 2021]



...

Ray tracing

- Early ray tracing
 - Whitted [1980]
 - Cook [1984]
- Path tracing
 - ...
 - ...
 - many recent papers

Rasterization vs. Ray Tracing



Rasterization

- OpenGL / DirectX
- Many techniques learned in undergrad-graphics courses

Rasterization finds intersections of a set of rays with the same origin/direction

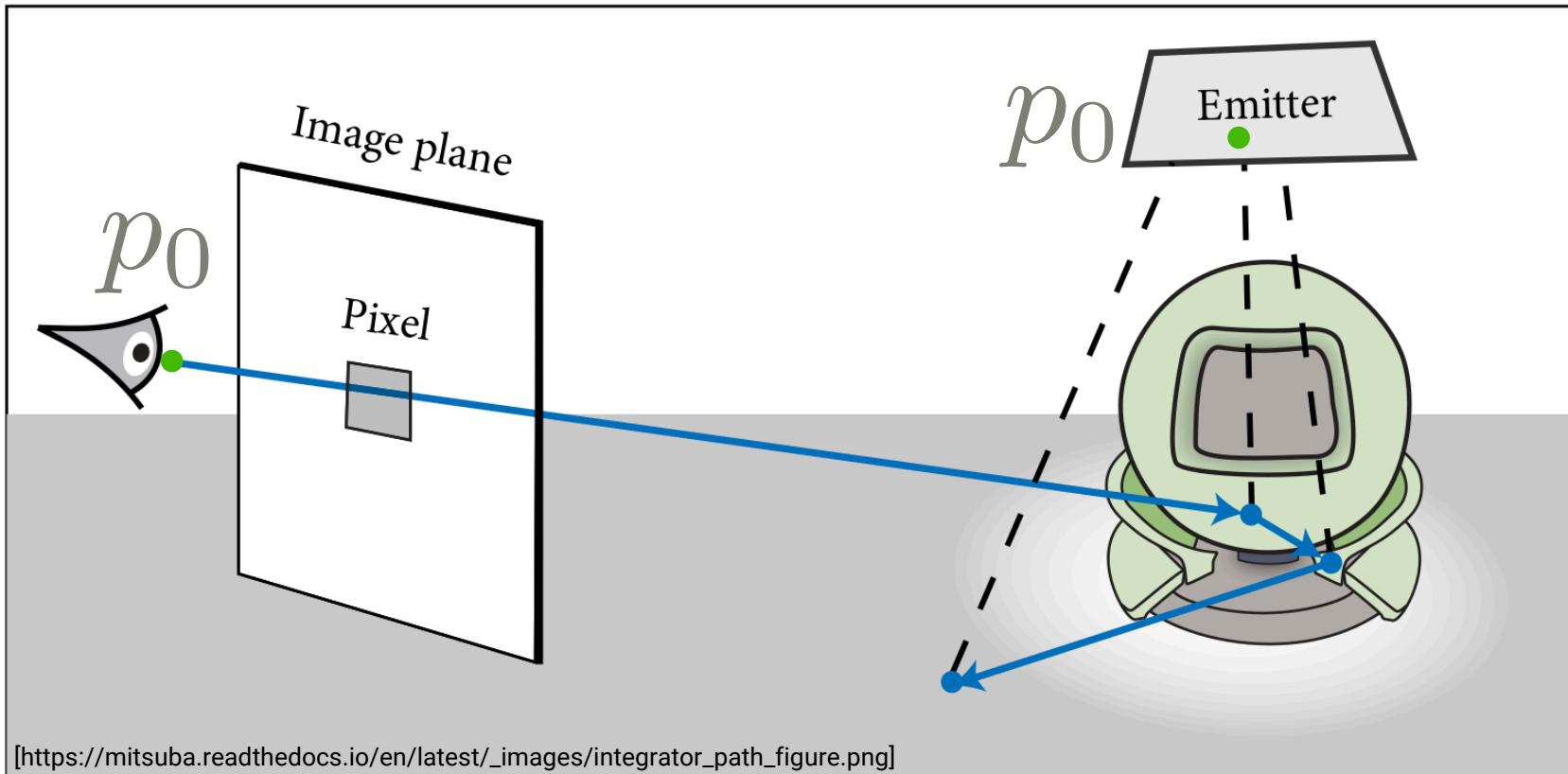
Finding a single ray intersection is inefficient

Ray tracing

- Early ray tracing
 - Whitted [1980]
 - Cook [1984]
- Path tracing
- ...
- ...
- many recent papers

Recursively perform the ray intersection query

Path Tracing: Overview



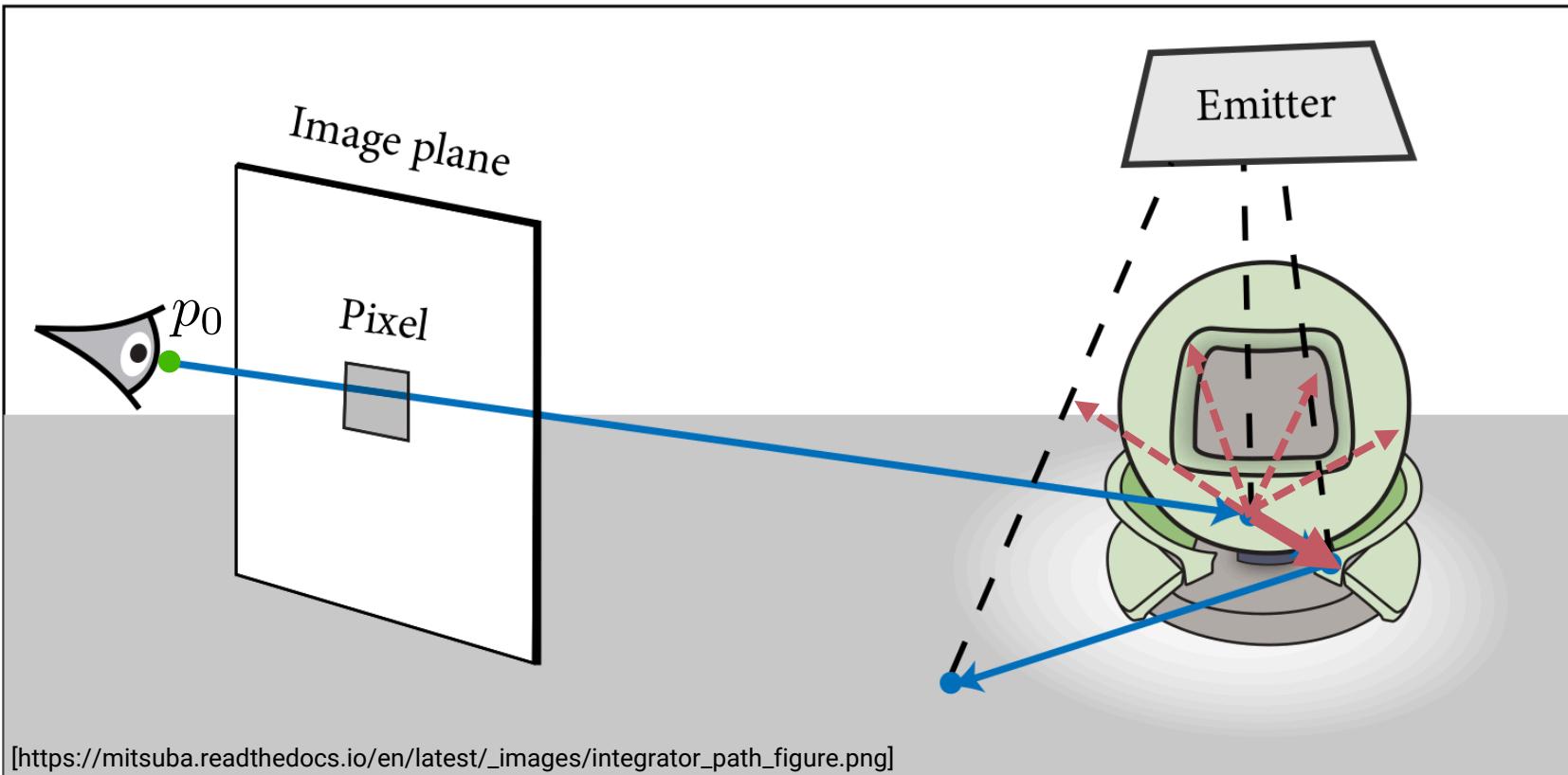
Q. Start from **sensor** vs. **emitter**?

(narrow meaning)
path tracing

light tracing =
particle tracing

→ **sensor**

Path Tracing: Overview



[https://mitsuba.readthedocs.io/en/latest/_images/integrator_path_figure.png]

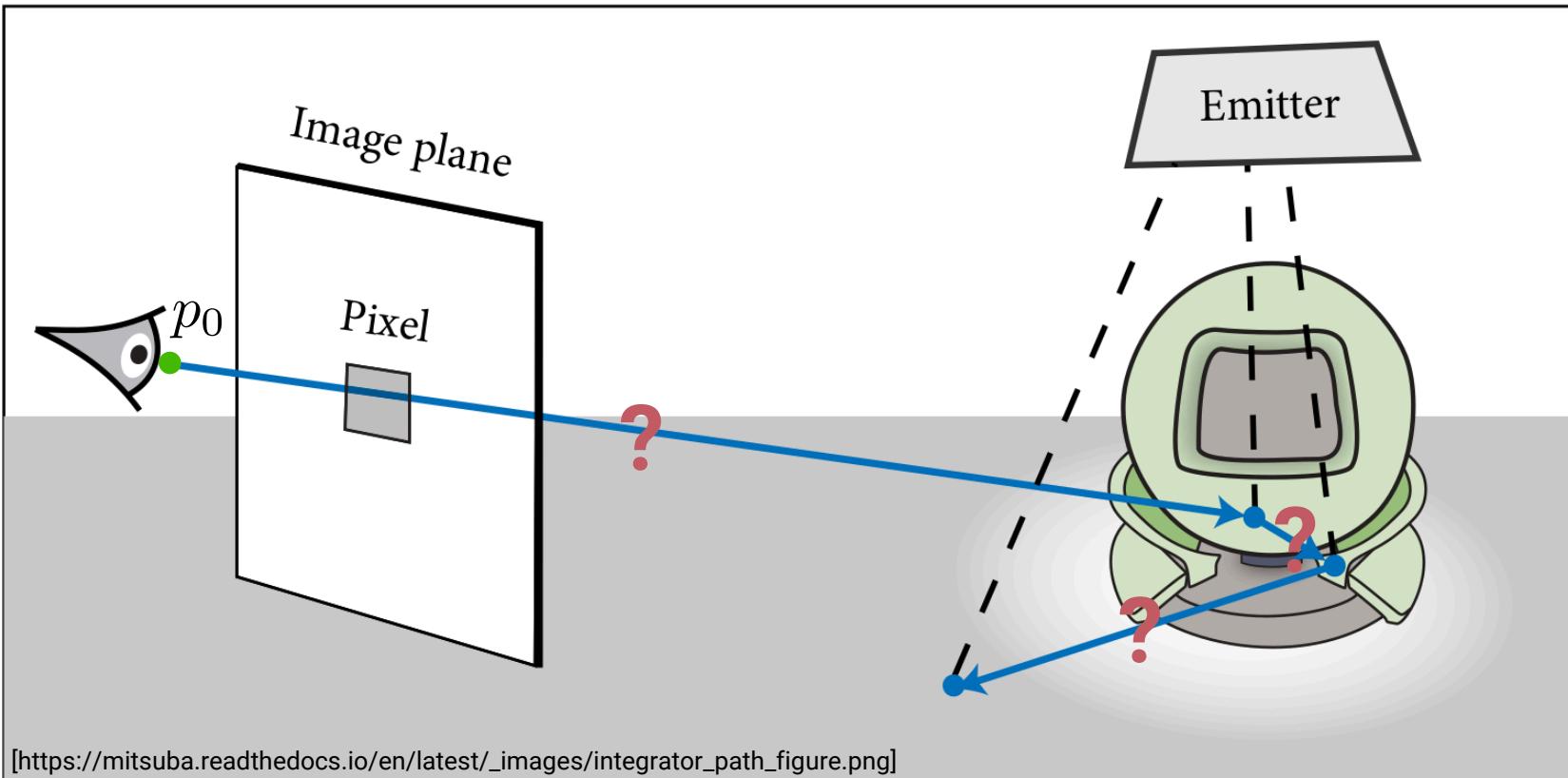
Q. Multiple rays vs. single ray for each bounce?

[Cook 1984] "Distributed ray tracing"

Path tracing

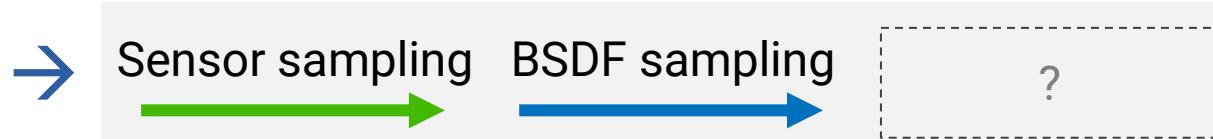
→ single ray

Path Tracing: Overview

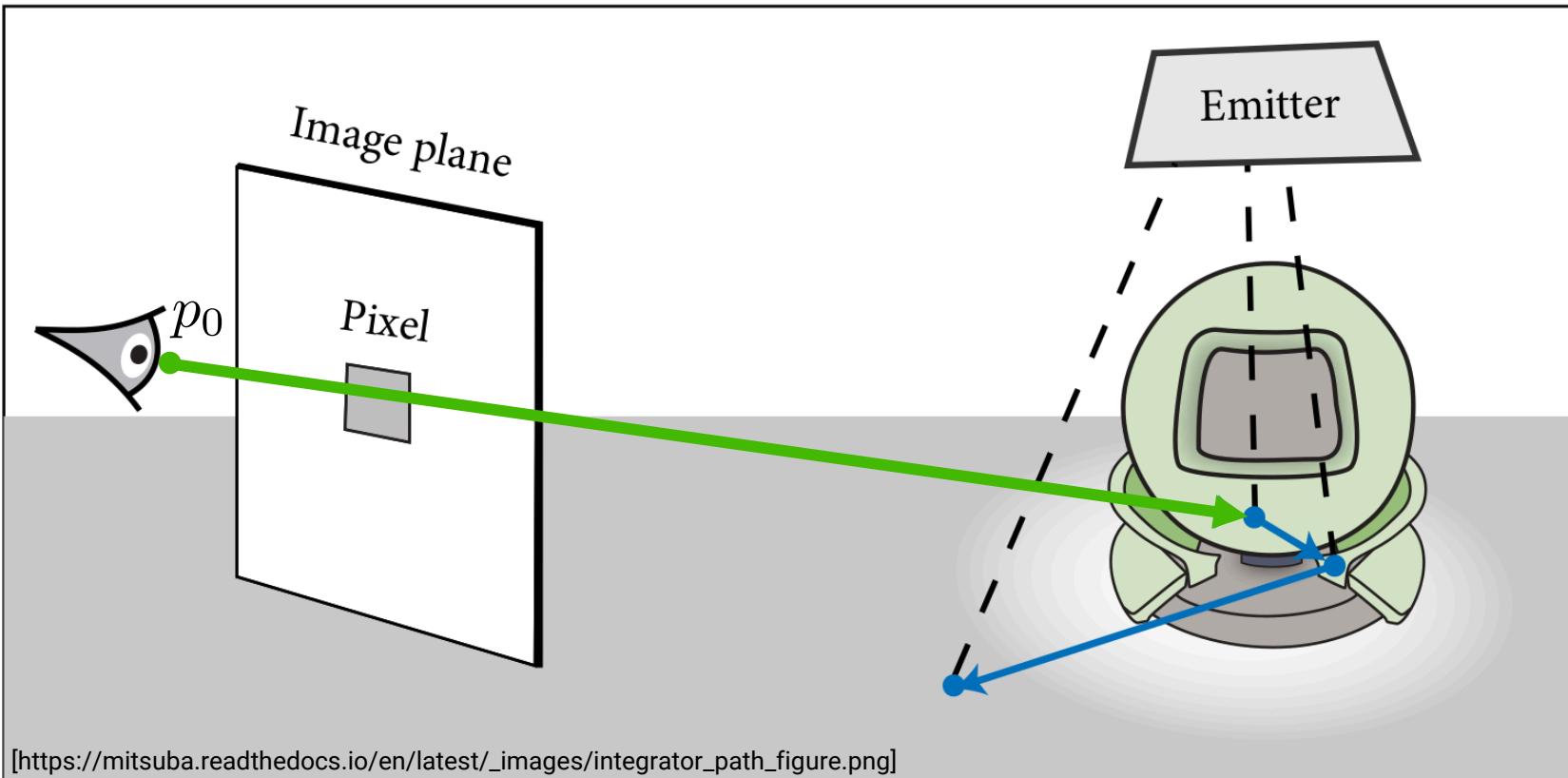


[https://mitsuba.readthedocs.io/en/latest/_images/integrator_path_figure.png]

Q. How to sample the next direction from given vertex?
= Which probability distribution



Path Tracing: Overview

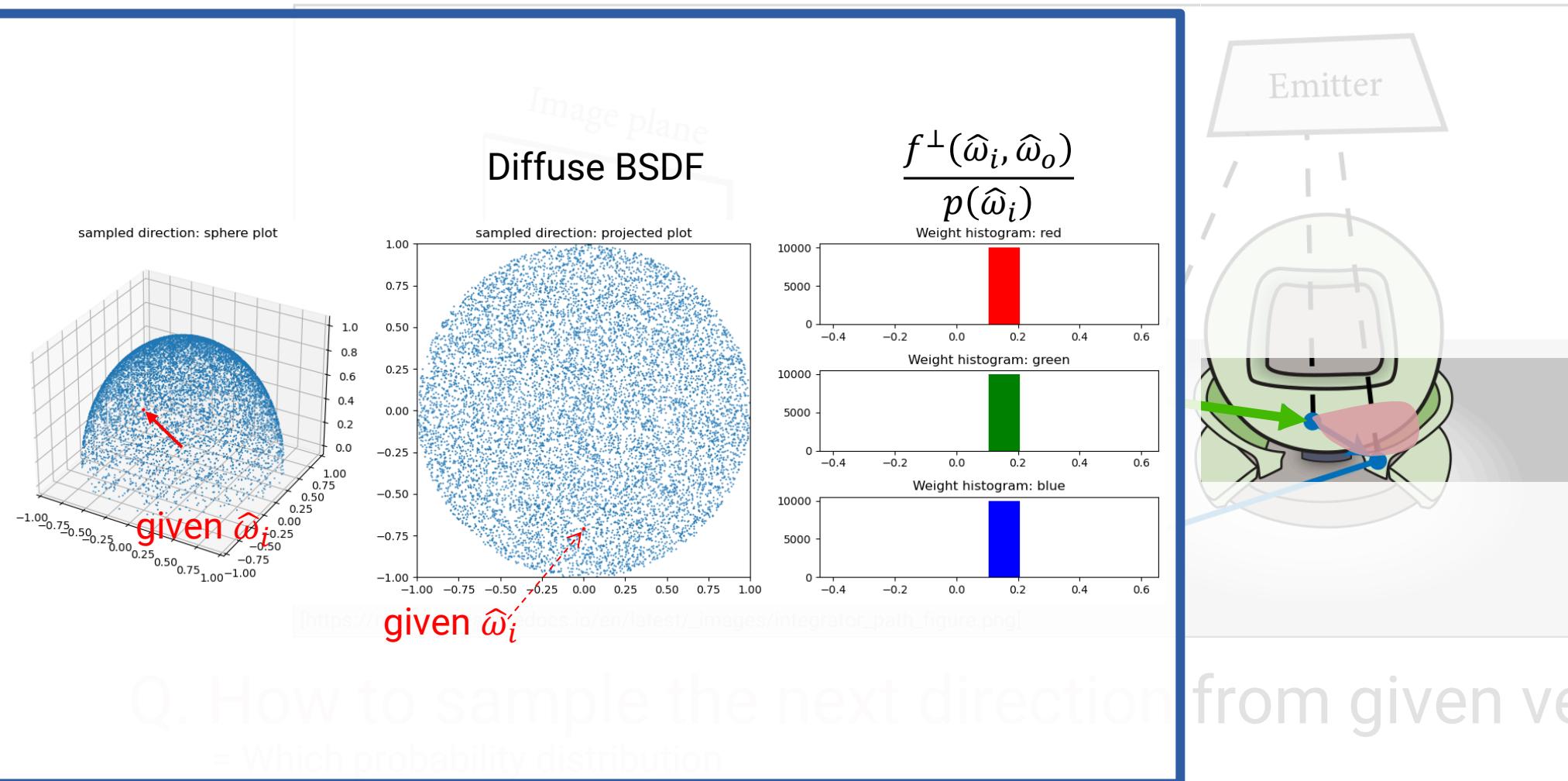


[https://mitsuba.readthedocs.io/en/latest/_images/integrator_path_figure.png]

Q. How to sample the next direction from given vertex?
= Which probability distribution



Path Tracing: Overview

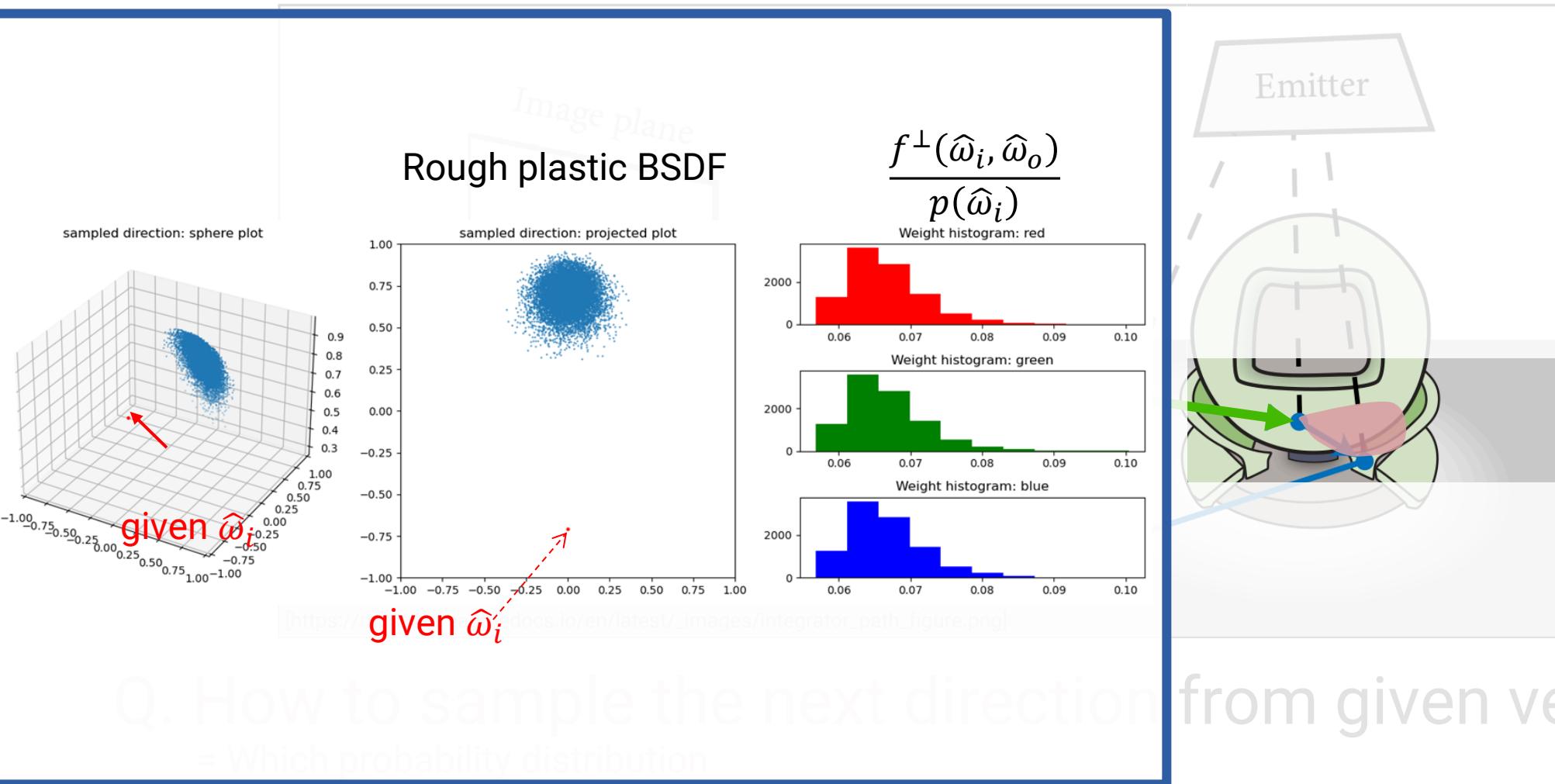


Sensor sampling

BSDF sampling

?

Path Tracing: Overview

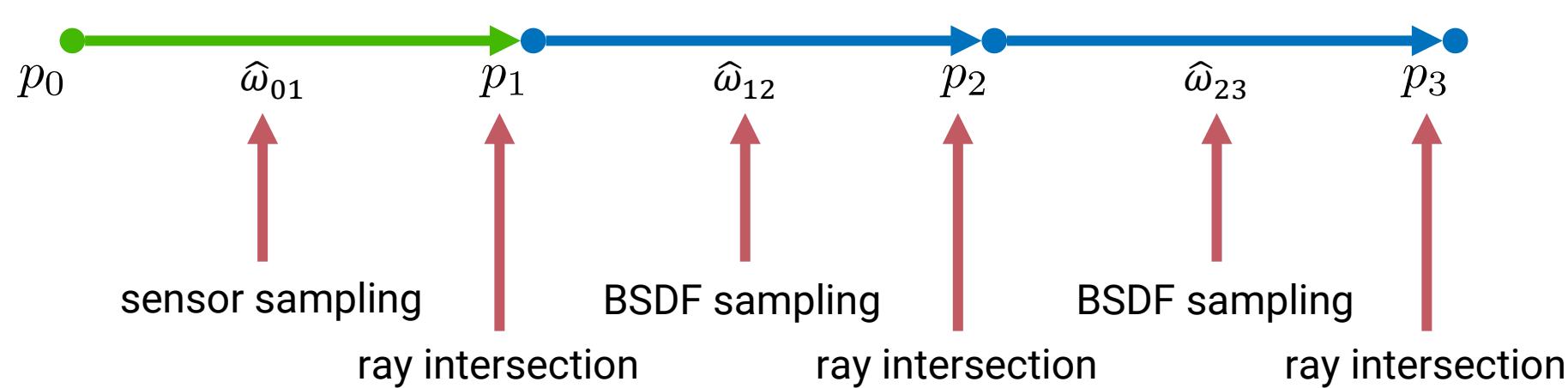


Sensor sampling

BSDF sampling

?

Path Tracing Overview



Rendering API

Ray intersection

Sensor sampling

BSDF sampling

?

Rendering & Measurement equation

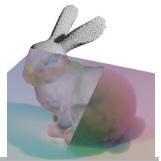


$$\begin{aligned} I = & \int_{\mathbb{S}^2} W_{e,10} L_{e,10} d\hat{\omega}_{01} \\ & + \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp L_{e,21} d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \\ & + \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp L_{e,32} d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \\ & + \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \end{aligned}$$

Monte Carlo!

$$\int_{\mathbb{S}^2} f(\hat{\omega}) d\hat{\omega} \rightarrow \frac{f(W)}{p(W)}$$

Rendering & Measurement equation



$$\hat{I} = \frac{W_{e,10} L_{e,10}}{p(W_{01})}$$
$$+ \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp L_{e,21} d\hat{\omega}_{12} \right] d\hat{\omega}_{01}$$
$$+ \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp L_{e,32} d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01}$$
$$+ \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01}$$

Monte Carlo!

$$\int_{\mathbb{S}^2} f(\hat{\omega}) d\hat{\omega} \rightarrow \frac{f(W)}{p(W)}$$

Rendering & Measurement equation



$$\begin{aligned}\hat{I} = & \frac{W_{e,10}L_{e,10}}{p(W_{01})} \\ & + \frac{W_{e,10}f_{s,210}^\perp L_{e,21}}{p(W_{01})p(W_{12})} \\ & + \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp L_{e,32} d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01} \\ & + \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01}\end{aligned}$$

Monte Carlo!

$$\int_{\mathbb{S}^2} f(\hat{\omega}) d\hat{\omega} \rightarrow \frac{f(W)}{p(W)}$$

Rendering & Measurement equation



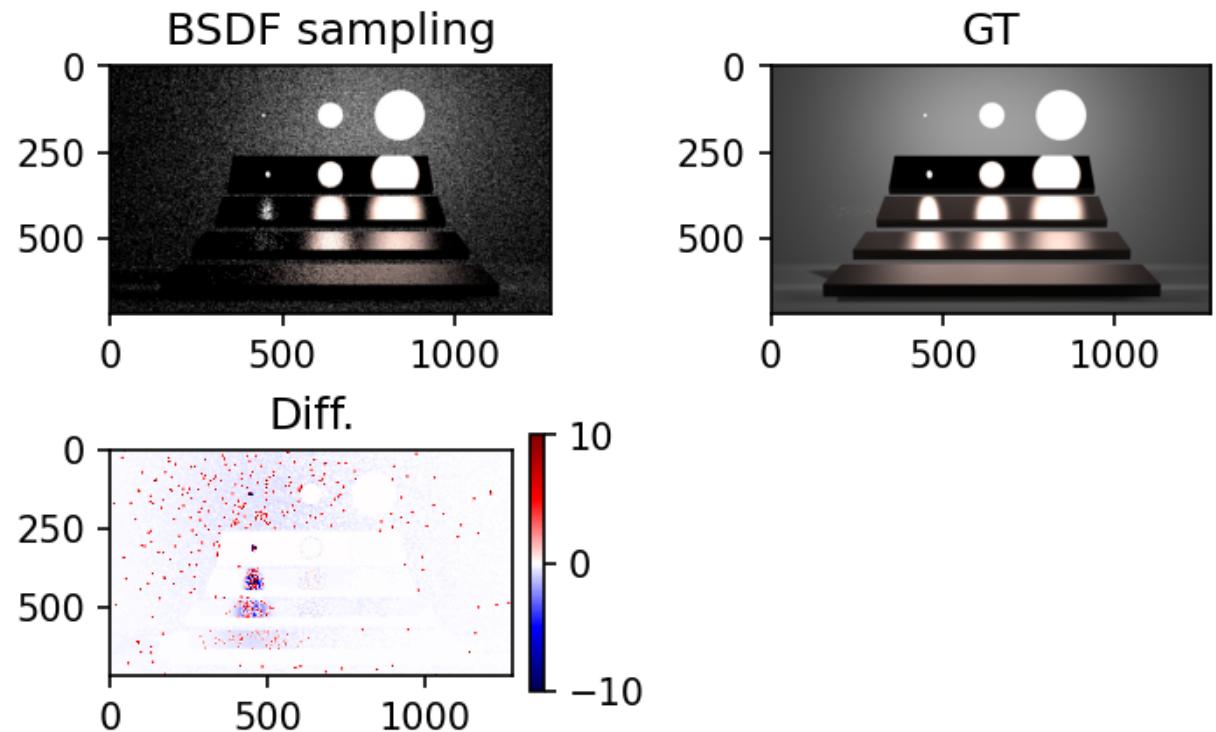
$$\begin{aligned}\hat{I} = & \frac{W_{e,10}L_{e,10}}{p(W_{01})} \\ & + \frac{W_{e,10}f_{s,210}^\perp L_{e,21}}{p(W_{01})p(W_{12})} \\ & + \frac{W_{e,10}f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p(W_{01})p(W_{12})p(W_{23})} \\ & + \int_{\mathbb{S}^2} W_{e,10} \left[\int_{\mathbb{S}^2} f_{s,210}^\perp \left[\int_{\mathbb{S}^2} f_{s,321}^\perp \left[\int_{\mathbb{S}^2} f_{s,432}^\perp L_{e,43} d\hat{\omega}_{34} \right] d\hat{\omega}_{23} \right] d\hat{\omega}_{12} \right] d\hat{\omega}_{01}\end{aligned}$$

Monte Carlo!

$$\int_{\mathbb{S}^2} f(\hat{\omega}) d\hat{\omega} \rightarrow \frac{f(W)}{p(W)}$$

BSDF Sampling

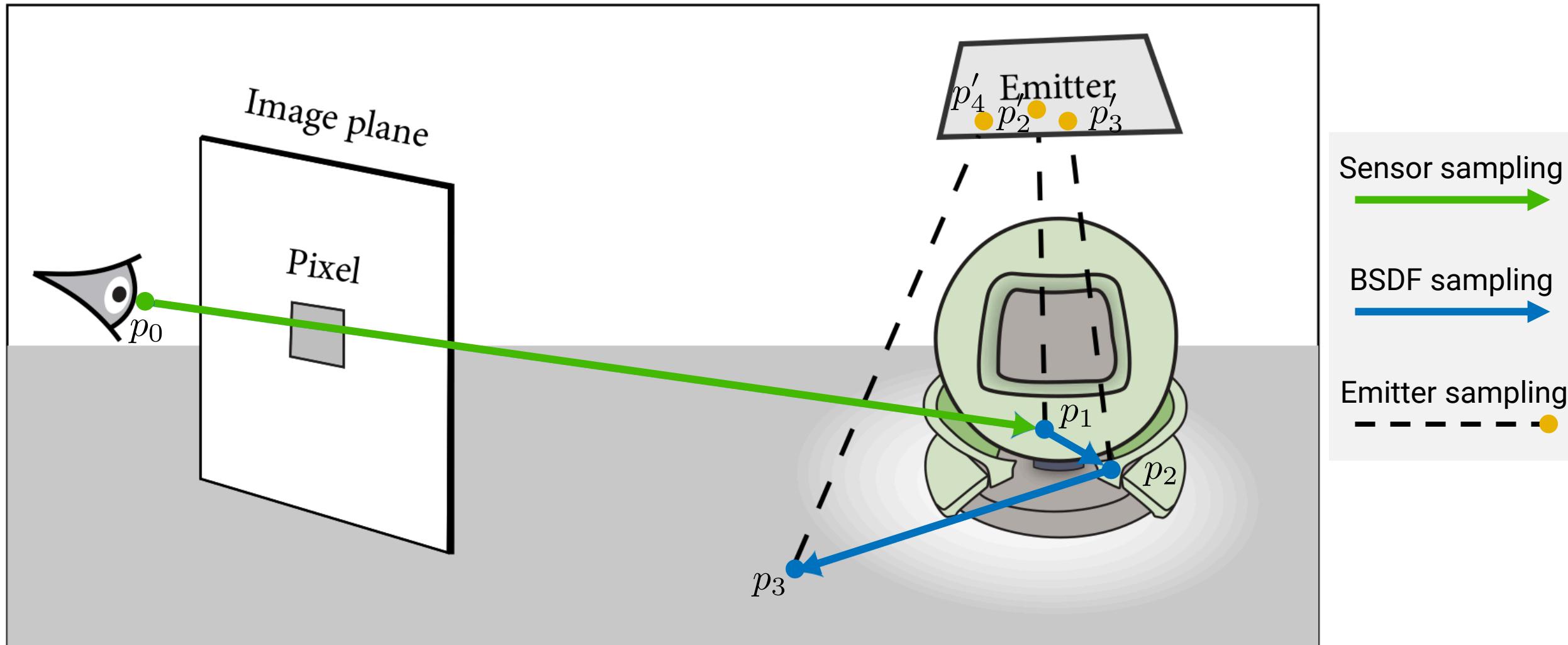
$$\hat{I} = \frac{W_{e,10}L_{e,10}}{p(W_{01})} + \frac{W_{e,10}f_{s,210}^\perp L_{e,21}}{p(W_{01})p(W_{12})} + \frac{W_{e,10}f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p(W_{01})p(W_{12})p(W_{23})} + \frac{W_{e,10}f_{s,210}^\perp f_{s,321}^\perp f_{s,432}^\perp L_{e,43}}{p(W_{01})p(W_{12})p(W_{23})p(W_{34})}$$



Monte Carlo!

$$\int_{\mathbb{S}^2} f(\hat{\omega}) d\hat{\omega} \rightarrow \frac{f(W)}{p(W)}$$

Emitter Sampling



[https://mitsuba.readthedocs.io/en/latest/_images/integrator_path_figure.png]

Rendering & Measurement equation



$$\hat{I} = \frac{W_{e,10}L_{e,10}}{p(W_{01})} + \frac{W_{e,10}f_{s,210}^\perp L_{e,21}}{p(W_{01})p(W_{12})} + \frac{W_{e,10}f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p(W_{01})p(W_{12})p(W_{23})} + \frac{W_{e,10}f_{s,210}^\perp f_{s,321}^\perp f_{s,432}^\perp L_{e,43}}{p(W_{01})p(W_{12})p(W_{23})p(W_{34})}$$

Rendering Equation



$$\begin{aligned}\hat{L} = & L_{e,10} \\ & + \frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})} \\ & + \frac{f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p_B(W_{12,B}) p_B(W_{23,B})} \\ & + \frac{f_{s,210}^\perp f_{s,321}^\perp f_{s,432}^\perp L_{e,43}}{p_B(W_{12,B}) p_B(W_{23,B}) p_B(W_{34,B})}\end{aligned}$$

BSDF sampling

Emitter sampling

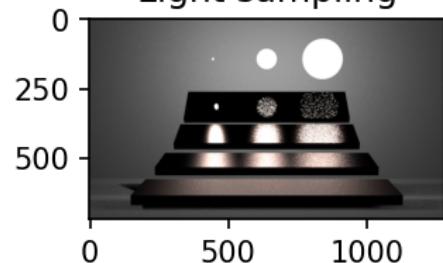
Rendering Equation with Emitter Sampling



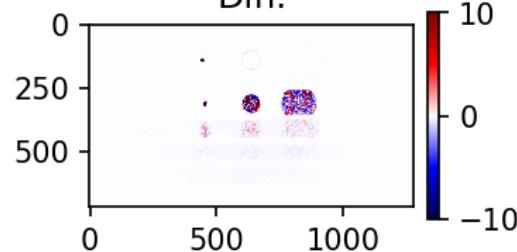
$$\hat{L} = L_{e,10}$$

$$+ \frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})}$$

Light sampling



Diff.



$$+ \frac{f_{s,210}^\perp L_{e,21}}{p_E(W_{12,E})}$$

$$+ \frac{f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p_B(W_{12,B}) p_E(W_{23,E})}$$

$$+ \frac{f_{s,210}^\perp f_{s,321}^\perp f_{s,432}^\perp L_{e,43}}{p_B(W_{12,B}) p_B(W_{23,B}) p_E(W_{34,E})}$$

BSDF sampling

Emitter sampling

Which Sampling Strategy?



$$\hat{L} = L_{e,10} + \frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})}$$

vs.

$$+ \frac{f_{s,210}^\perp L_{e,21}}{p_E(W_{12,E})}$$

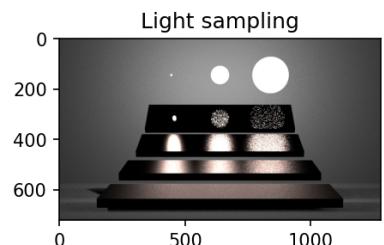
BSDF sampling

Emitter sampling

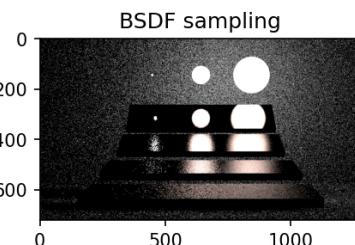
Which Sampling Strategy?



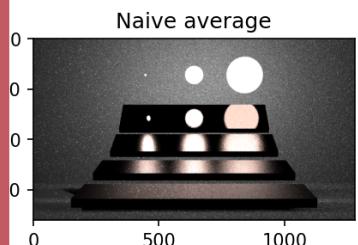
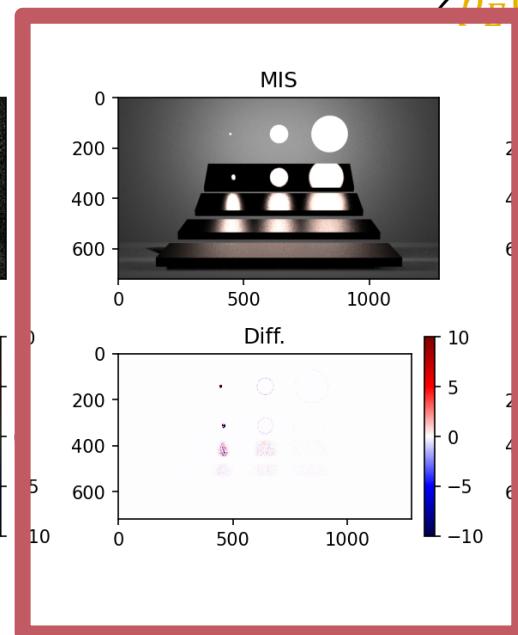
$$\frac{f_{s,210}^\perp L_{e,21}}{p_E(W_{12,E})}$$



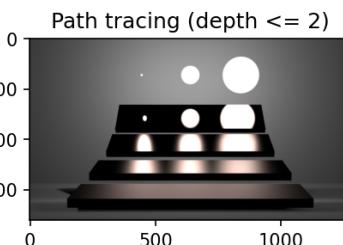
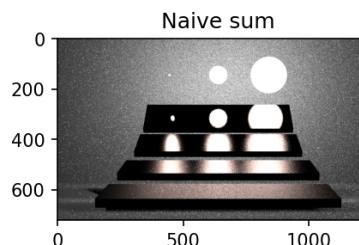
$$\frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})}$$



$$\frac{f_{s,210}^\perp L_{e,21}}{2p_E(W_{12,E})} + \frac{f_{s,210}^\perp L_{e,21}}{2p_B(W_{12,B})}$$

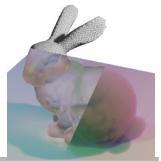


$$\frac{f_{s,210}^\perp L_{e,21}}{p_E(W_{12,E})} + \frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})}$$



Multiple Importance Sampling (MIS)

MIS for BSDF and Emitter Sampling Methods



$$\hat{L} = L_{e,10}$$

$$\begin{aligned} & w_B(W_{12,B}) \frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})} + w_E(W_{12,E}) \frac{f_{s,210}^\perp L_{e,21}}{p_E(W_{12,E})} \\ & w_B(W_{23,B}) \frac{f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p_B(W_{12,B}) p_B(W_{23,B})} + w_E(W_{23,E}) \frac{f_{s,210}^\perp f_{s,321}^\perp L_{e,32}}{p_B(W_{12,B}) p_E(W_{23,E})} \\ & + w_B(W_{34,B}) \frac{f_{s,210}^\perp f_{s,321}^\perp f_{s,432}^\perp L_{e,43}}{p_B(W_{12,B}) p_B(W_{23,B}) p_B(W_{34,B})} + w_E(W_{34,E}) \frac{f_{s,210}^\perp f_{s,321}^\perp f_{s,432}^\perp L_{e,43}}{p_B(W_{12,B}) p_B(W_{23,B}) p_E(W_{34,E})} \end{aligned}$$

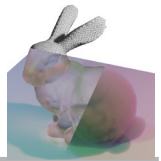
BSDF sampling

Emitter sampling

MIS weights

(See the full version slide from our repo)

MIS for BSDF and Emitter Sampling Methods



$$\hat{L} = L_{e,10}$$

$$\begin{aligned} & + w_B(W_{12,B}) \frac{f_{s,210}^\perp L_{e,21}}{p_B(W_{12,B})} + w_E(W_{12,E}) \frac{f_{s,210}^\perp L_{e,21}}{p_E(W_{12,E})} \\ & + \frac{f_{s,210}^\perp}{p_B(W_{12,B})} \left[w_B(W_{23,B}) \frac{f_{s,321}^\perp L_{e,32}}{p_B(W_{23,B})} + w_E(W_{23,E}) \frac{f_{s,321}^\perp L_{e,32}}{p_E(W_{23,E})} \right] \\ & + \frac{f_{s,210}^\perp f_{s,321}^\perp}{p_B(W_{12,B}) p_B(W_{23,B})} \left[w_B(W_{34,B}) \frac{f_{s,432}^\perp L_{e,43}}{p_B(W_{34,B})} + w_E(W_{34,E}) \frac{f_{s,432}^\perp L_{e,43}}{p_E(W_{34,E})} \right] \end{aligned}$$

throughput variable β

BSDF sampling

Emitter sampling

Russian roulette

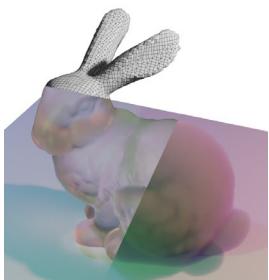


finite computation infinite sum???

$$\sum_{i=1}^{\infty} a_i$$

$$A_i, \mathbb{E}[A_i] = a_i$$

$$var \leftarrow var + \begin{cases} \frac{A_i}{p(\quad)} \\ 0, \end{cases}$$



Thank you

<https://syyi.graphics>

syyi.graphics@gmail.com

<https://github.com/shinyoung-yi/lecture-rendering-mitsuba>

Feel free to contact to me!