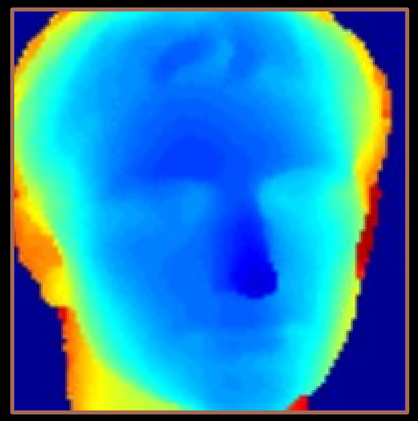
Physics-based rendering and its applications in computational photography and imaging

Adithya Pediredla
Dartmouth College
adithya.k.pediredla@dartmouth.com

Ioannis Gkioulekas
Carnegie Mellon University
igkioule@andrew.cmu.edu

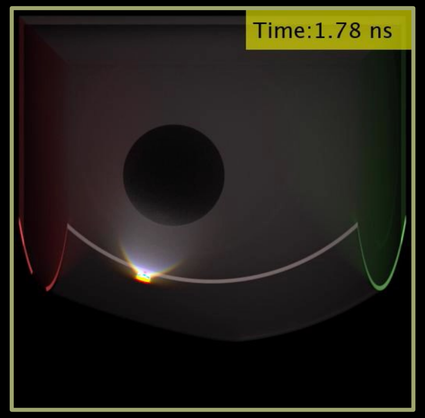
computer vision

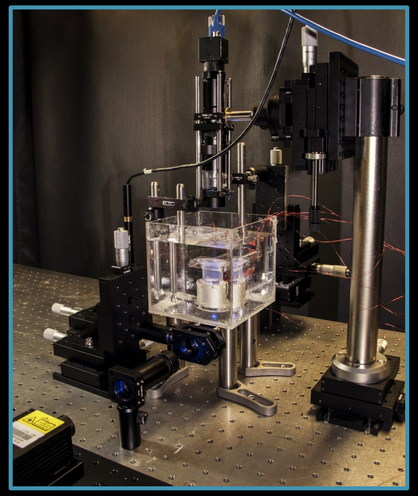


geometry
features
learning
optical flow

rendering
light transport
materials
AR/VR

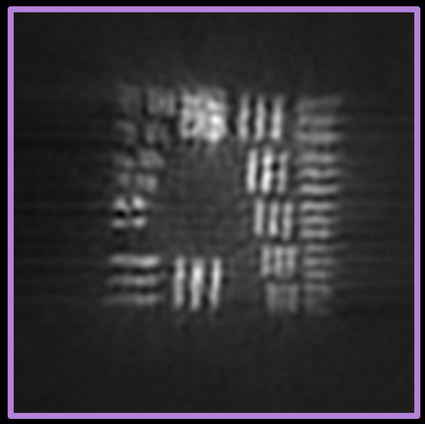
computer graphics





lens design
wave optics
non-linear optics
quantum optics

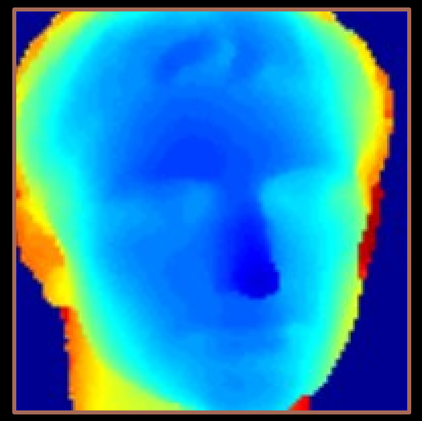
transforms
filtering
compression
linear models



optics

image processing

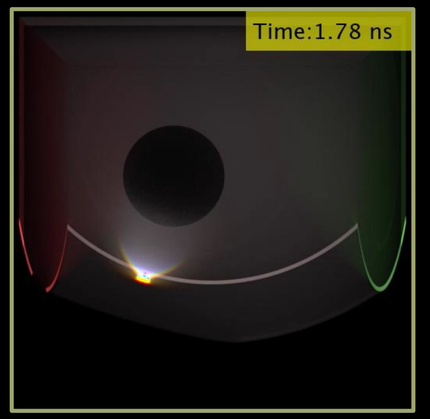
computer vision



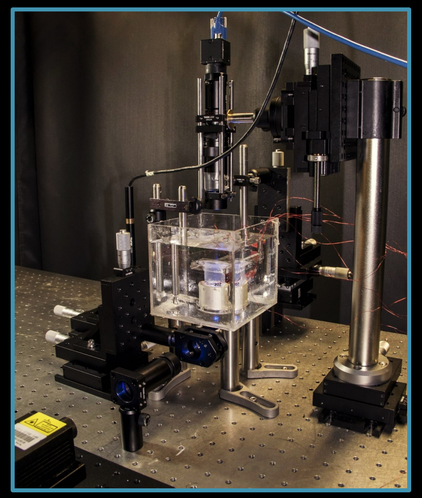
geometry
features
learning
optical flow

rendering
light transport
materials
AR/VR

computer graphics

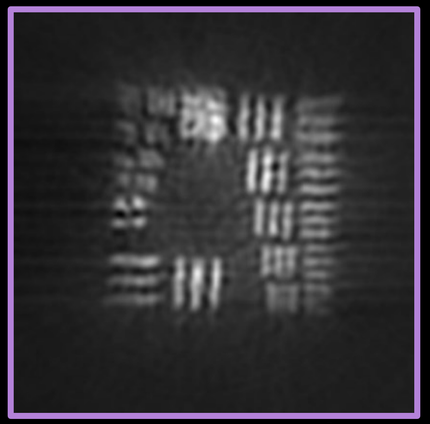


computational imaging



lens design
wave optics
non-linear optics
quantum optics

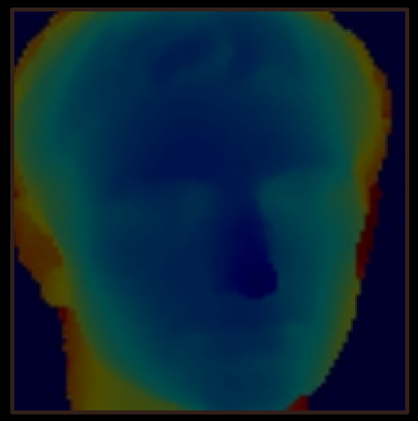
transforms
filtering
compression
linear models



optics

image processing

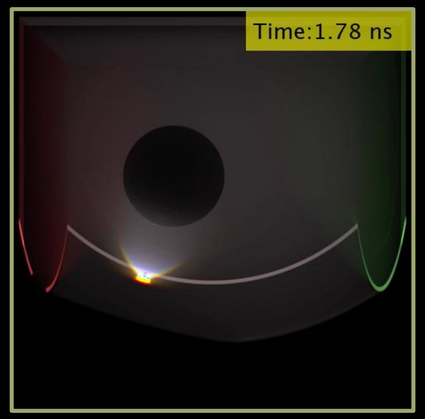
computer vision



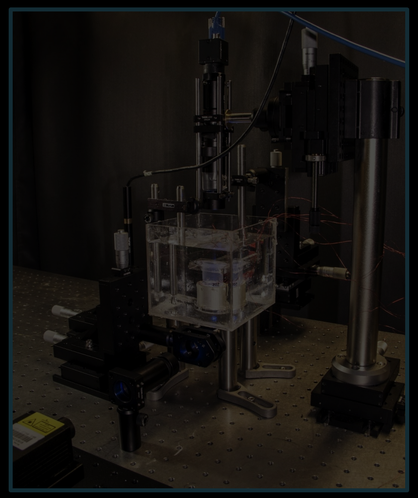
geometry
features
learning
optical flow

rendering
light transport
materials
AR/VR

computer graphics



computational imaging



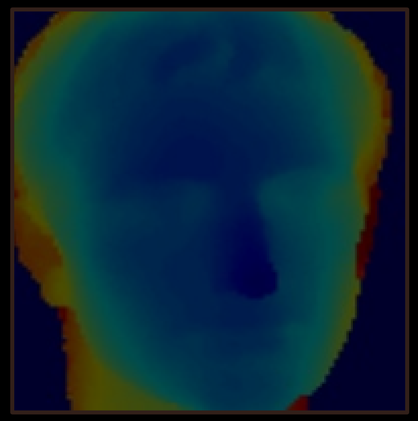
lens design
wave optics
non-linear optics
quantum optics

transforms
filtering
compression
linear models

optics

image processing

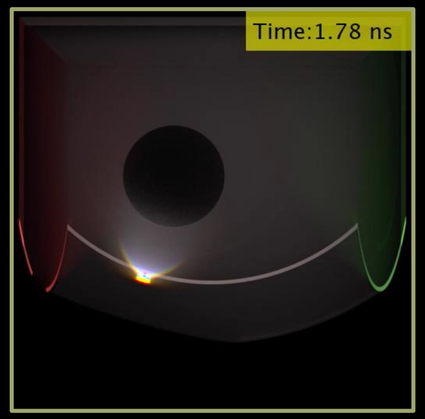
computer vision



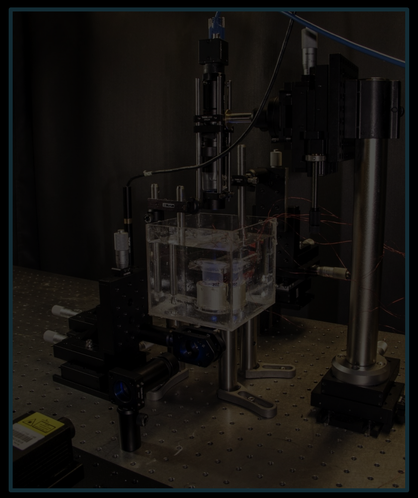
geometry
features
learning
optical flow

rendering
light transport
materials
AR/VR

computer graphics



computational imaging



lens design
wave optics
non-linear optics
quantum optics

transforms
filtering
compression
linear models

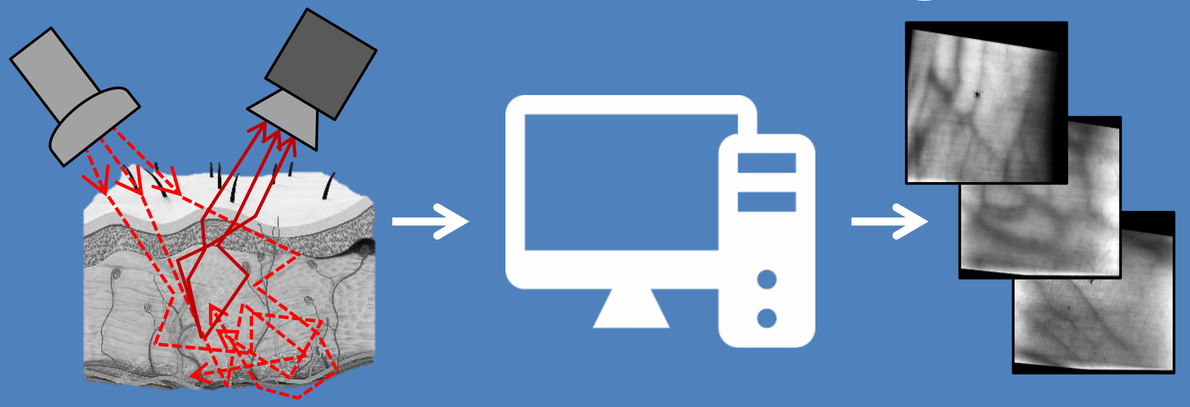
optics

image processing

Physics-based rendering and its applications to computational imaging

Physics-based rendering and its applications to computational imaging

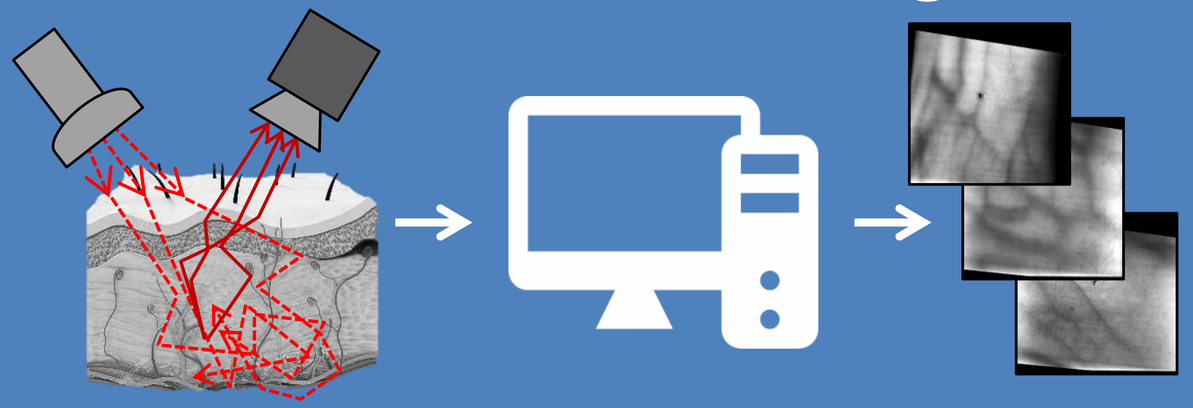
forward rendering



- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

Physics-based rendering and its applications to computational imaging

forward rendering



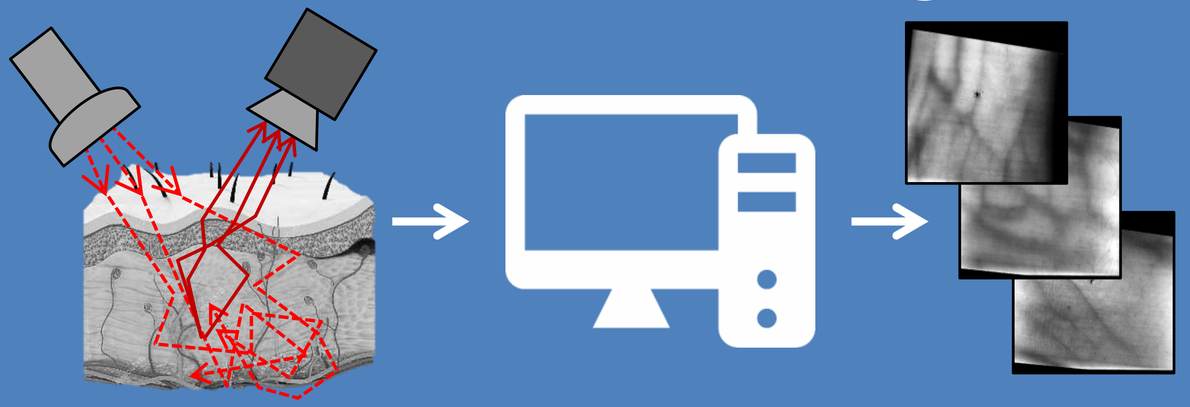
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

- accurate and efficient differentiable simulation
- tractably solve general inverse problems

Physics-based rendering and its applications to computational imaging

forward rendering



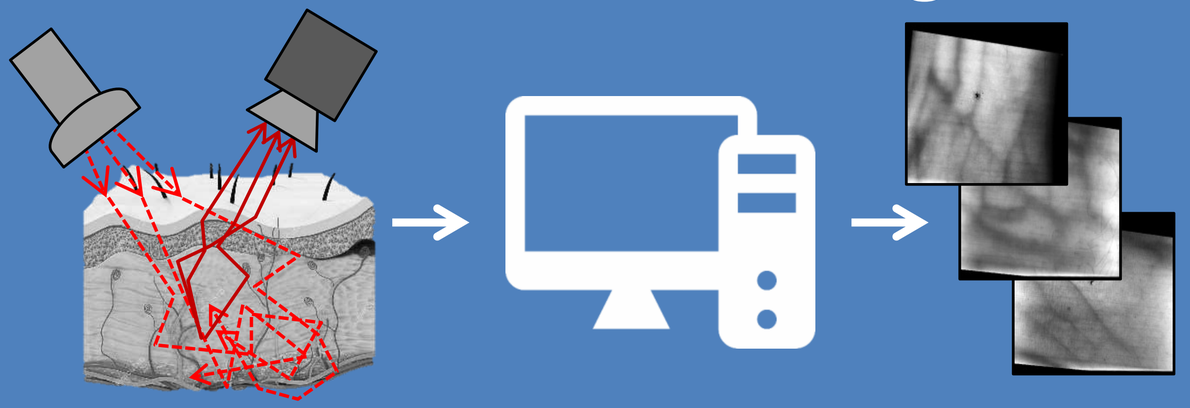
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

- accurate and efficient differentiable simulation
- tractably solve general inverse problems

Physics-based rendering and its applications to computational imaging

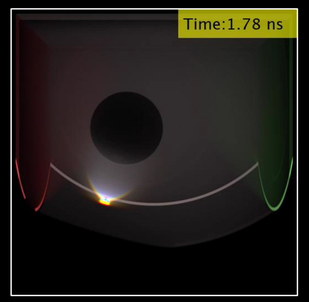
forward rendering



- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

- accurate and efficient differentiable simulation
- tractably solve general inverse problems



time-of-flight imaging

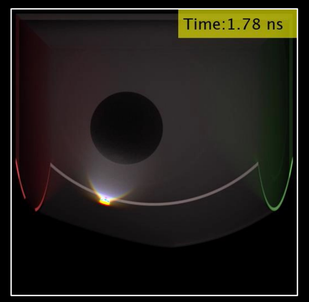
Physics-based rendering and its applications to computational imaging

forward rendering

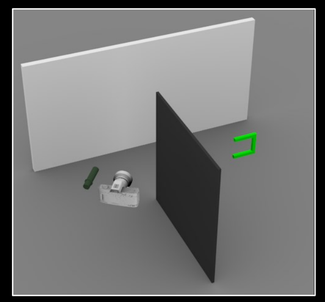
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

- accurate and efficient differentiable simulation
- tractably solve general inverse problems



time-of-flight imaging



non-line-of-sight imaging

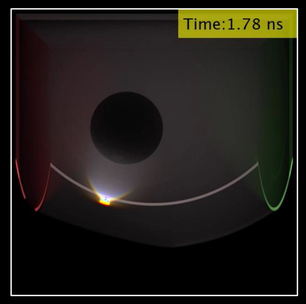
Physics-based rendering and its applications to computational imaging

forward rendering

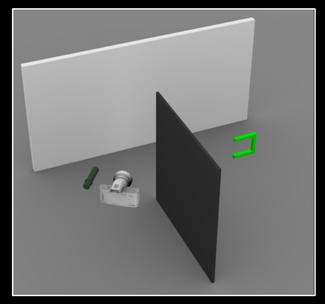
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

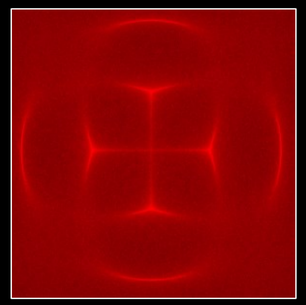
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



time-of-flight imaging



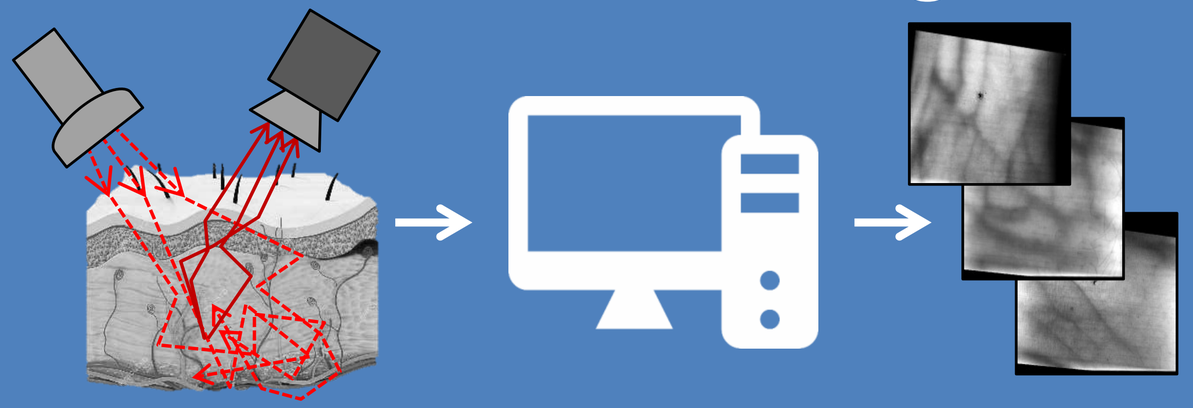
non-line-of-sight imaging



acousto-optic lensing

Physics-based rendering and its applications to computational imaging

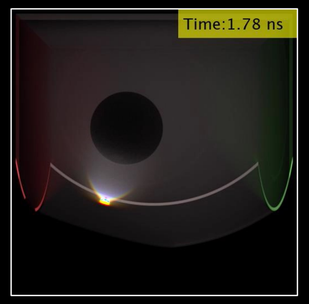
forward rendering



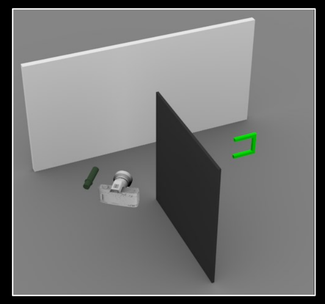
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

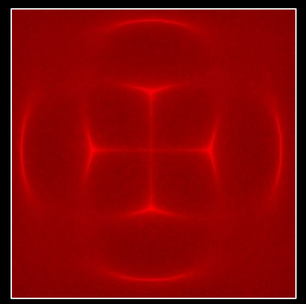
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



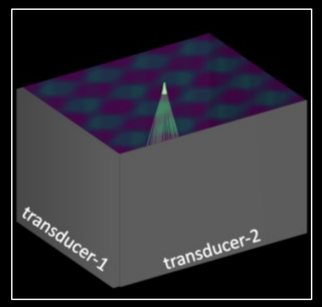
time-of-flight imaging



non-line-of-sight imaging



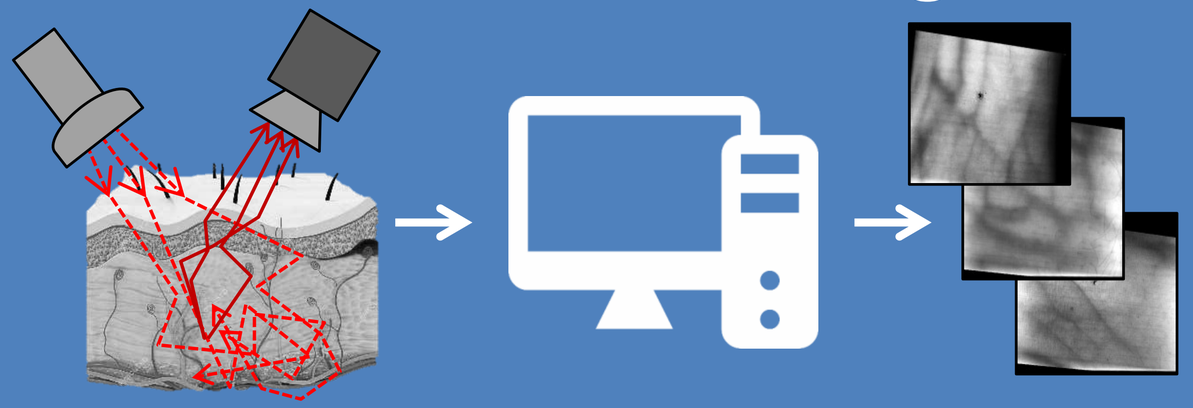
acousto-optic lensing



ultrafast light scanning

Physics-based rendering and its applications to computational imaging

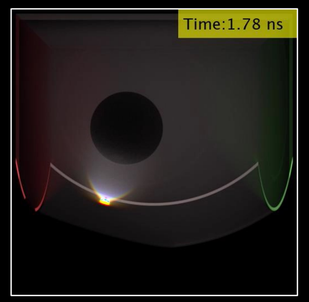
forward rendering



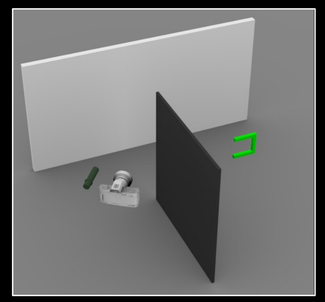
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

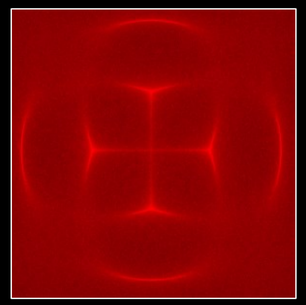
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



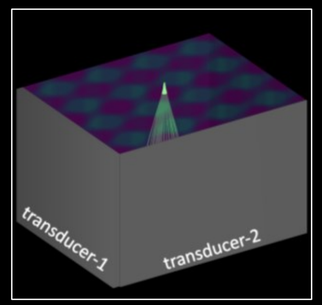
time-of-flight imaging



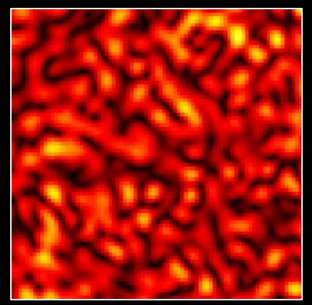
non-line-of-sight imaging



acousto-optic lensing



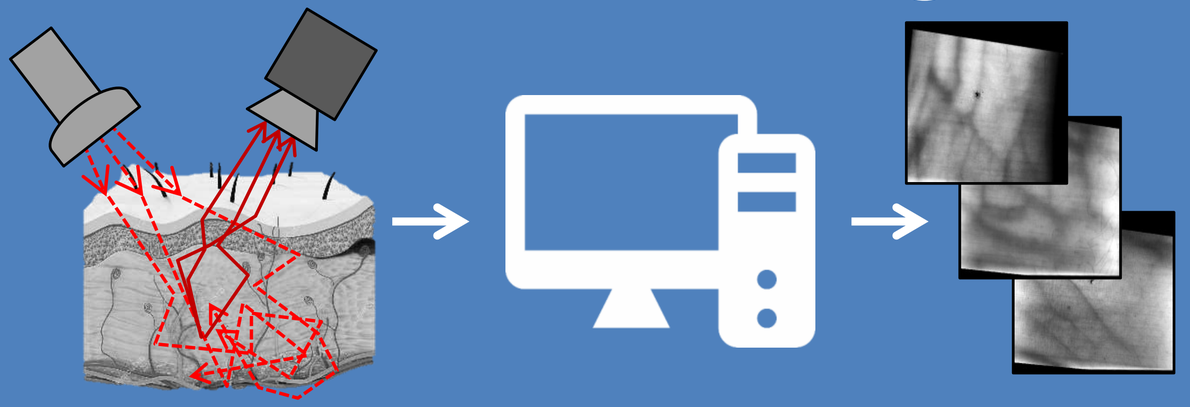
ultrafast light scanning



speckle imaging

Physics-based rendering and its applications to computational imaging

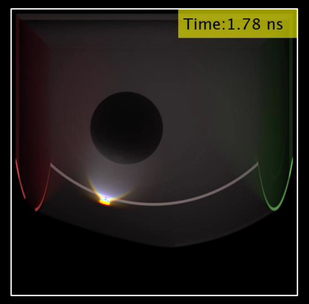
forward rendering



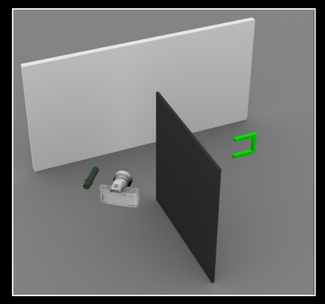
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

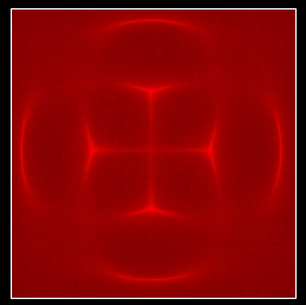
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



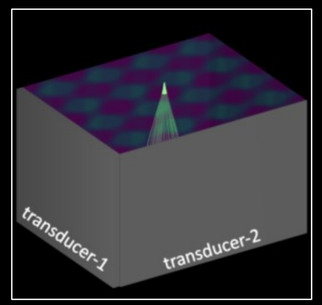
time-of-flight imaging



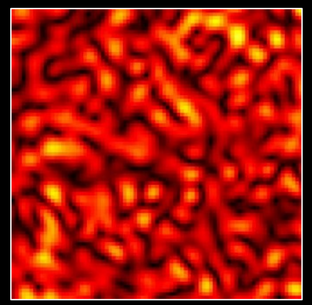
non-line-of-sight imaging



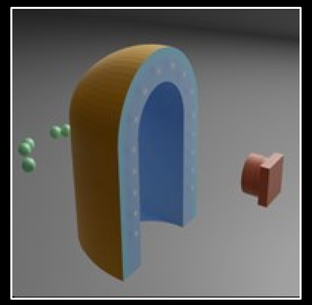
acousto-optic lensing



ultrafast light scanning



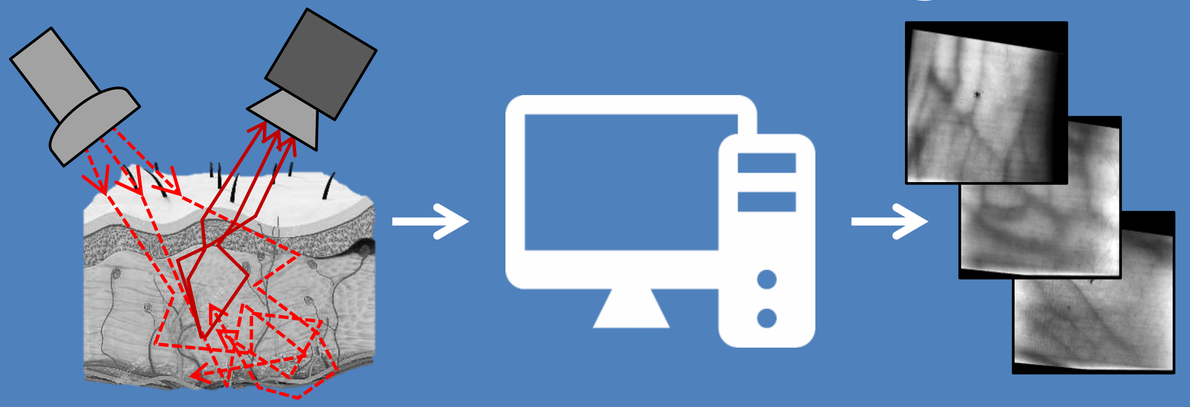
speckle imaging



tactile sensor design

Physics-based rendering and its applications to computational imaging

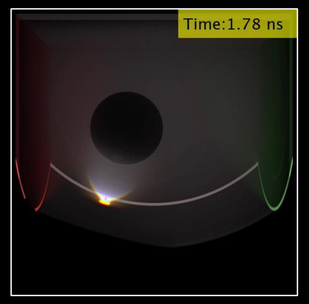
forward rendering



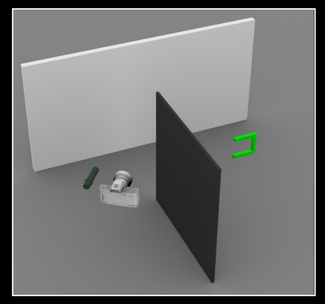
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

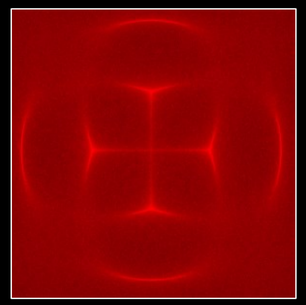
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



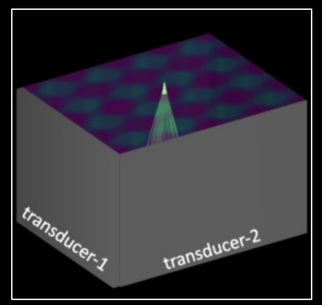
time-of-flight imaging



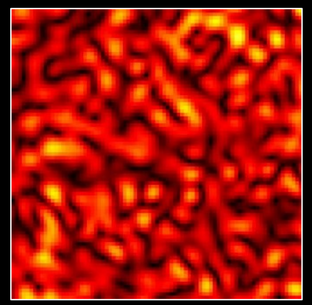
non-line-of-sight imaging



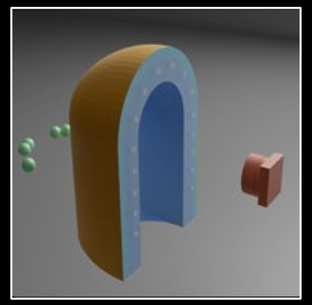
acousto-optic lensing



ultrafast light scanning



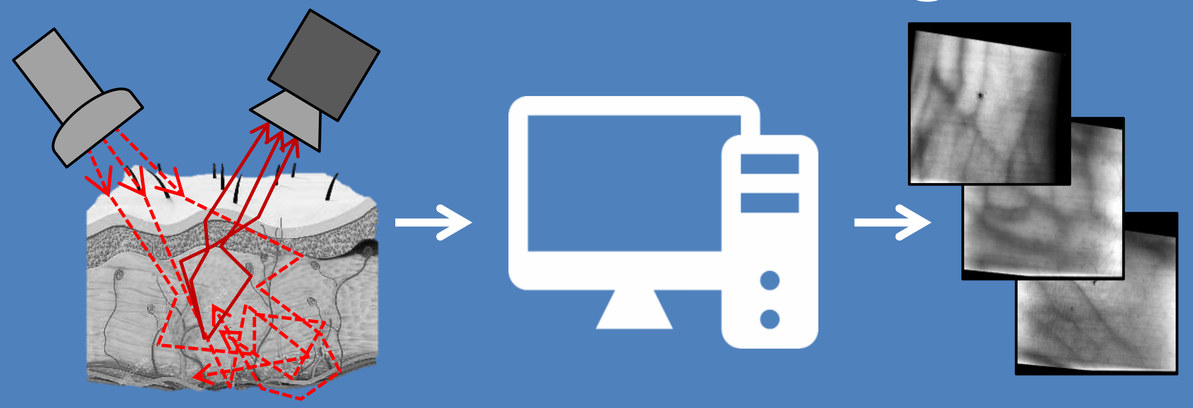
speckle imaging



tactile sensor design

Physics-based rendering and its applications to computational imaging

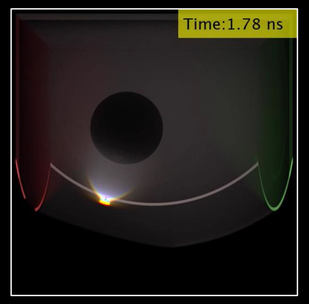
forward rendering



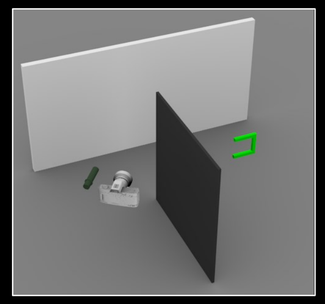
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

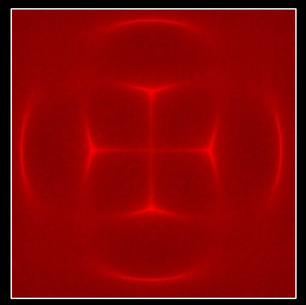
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



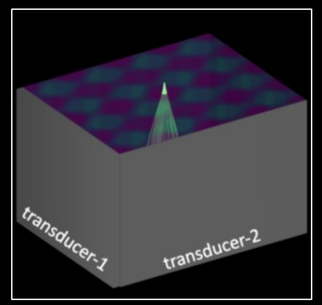
time-of-flight imaging



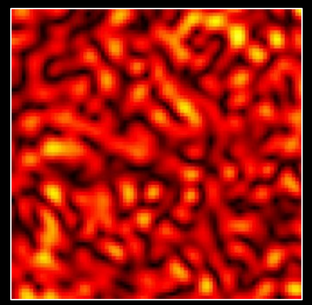
non-line-of-sight imaging



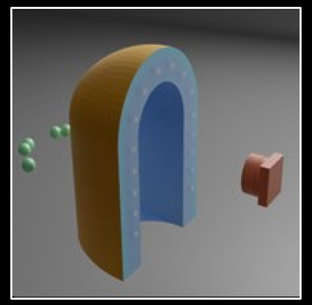
acousto-optic lensing



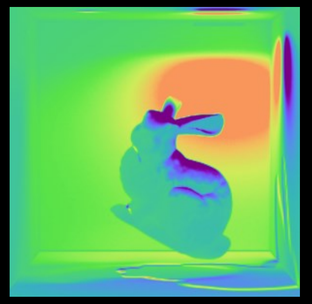
ultrafast light scanning



speckle imaging



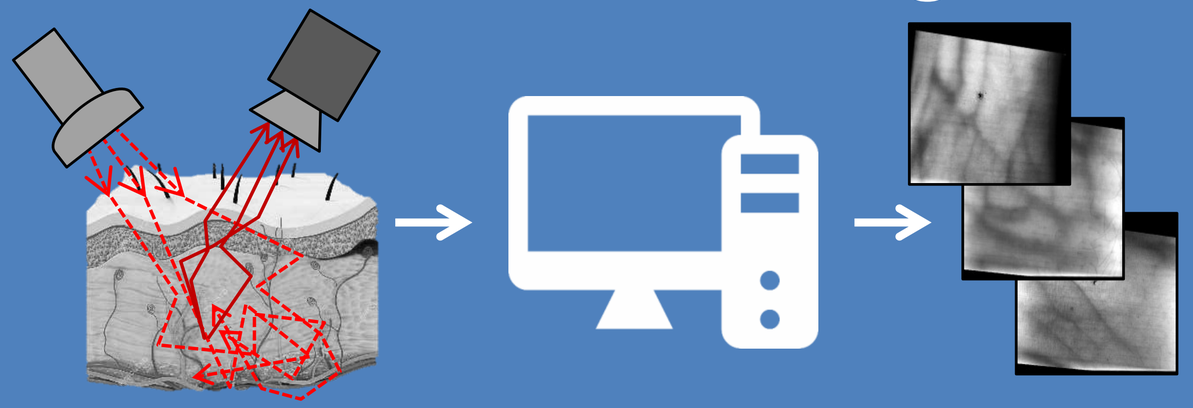
tactile sensor design



differentiable rendering

Physics-based rendering and its applications to computational imaging

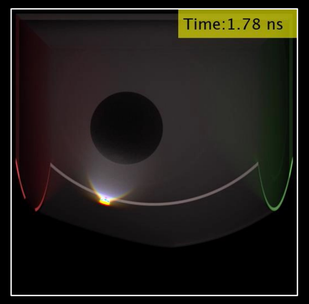
forward rendering



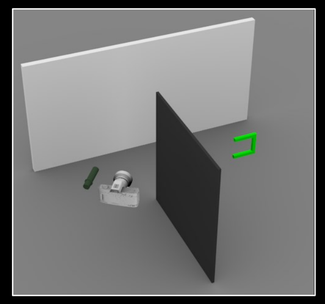
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

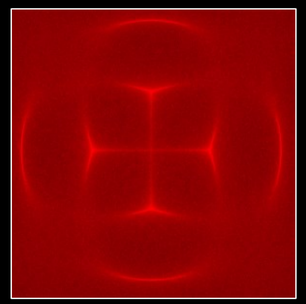
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



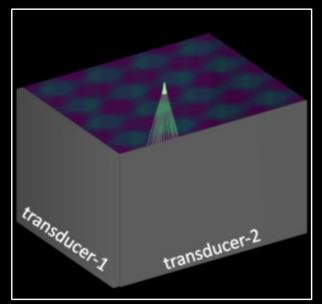
time-of-flight imaging



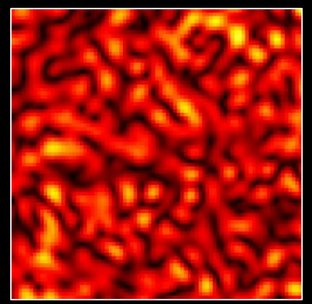
non-line-of-sight imaging



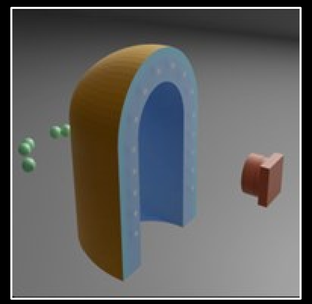
acousto-optic lensing



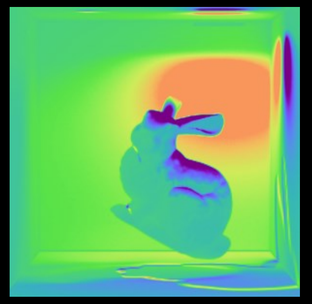
ultrafast light scanning



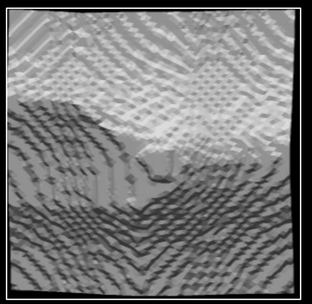
speckle imaging



tactile sensor design



differentiable rendering



inverse problems

Complex light transport

After [Ritschel et al 2011]

Complex light transport

After [Ritschel et al 2011]

Complex light transport

After [Ritschel et al 2011]

Complex light transport

After [Ritschel et al 2011]

Complex light transport

After [Ritschel et al 2011]

Complex light transport

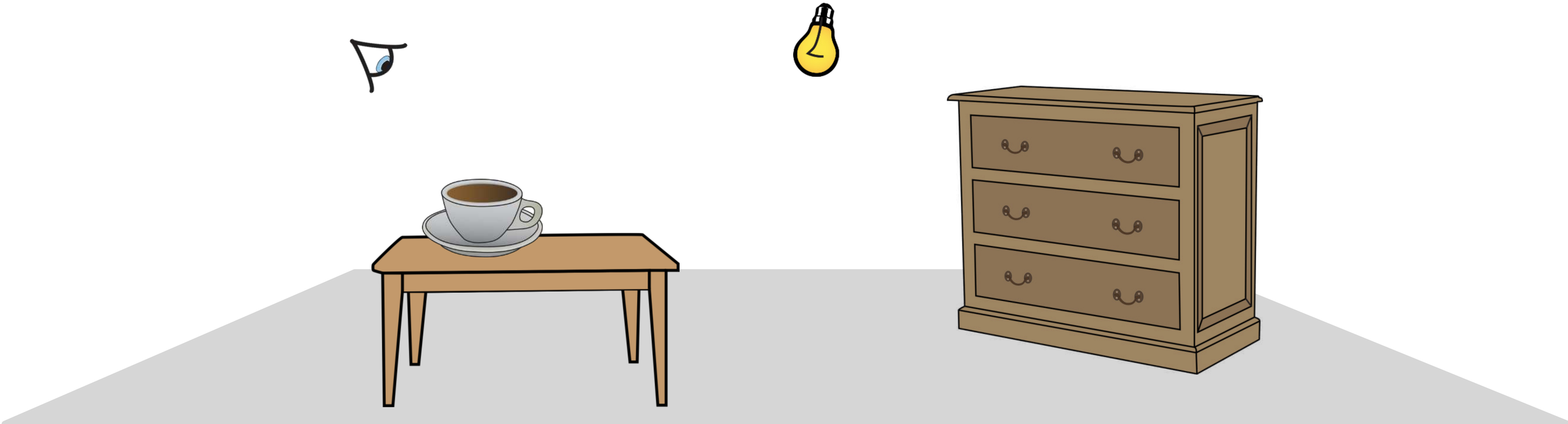
After [Ritschel et al 2011]

Complex light transport

After [Ritschel et al 2011]

Path integral form of light transport

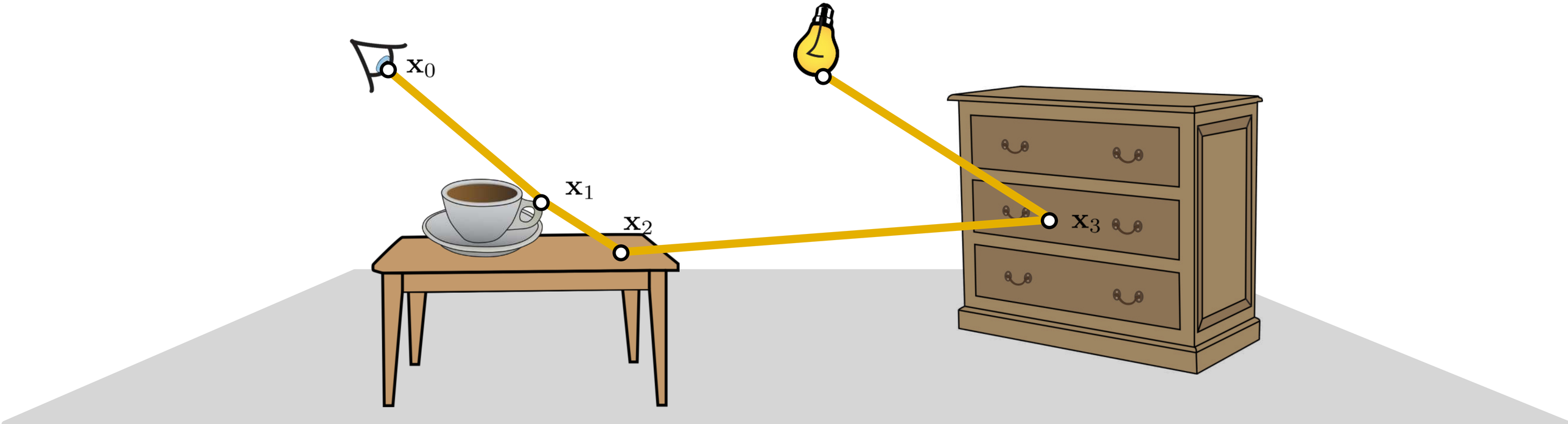
image $$I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$$



Path integral form of light transport

image $$I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$$ light path

space of all light paths



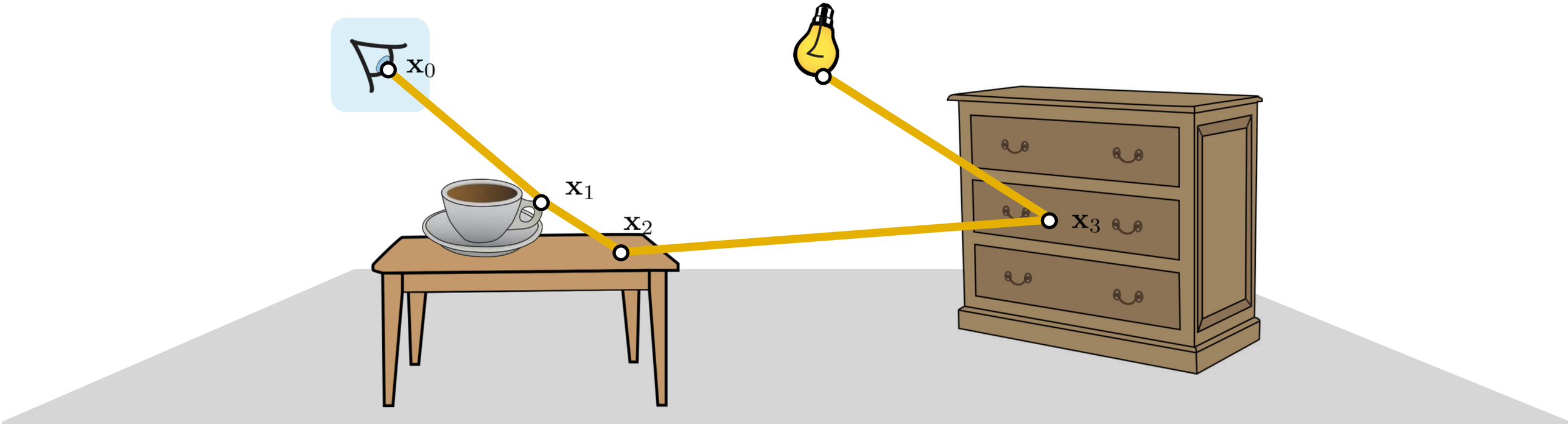
Path integral form of light transport

sensor

weight

image $I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$ light path

space of all light paths



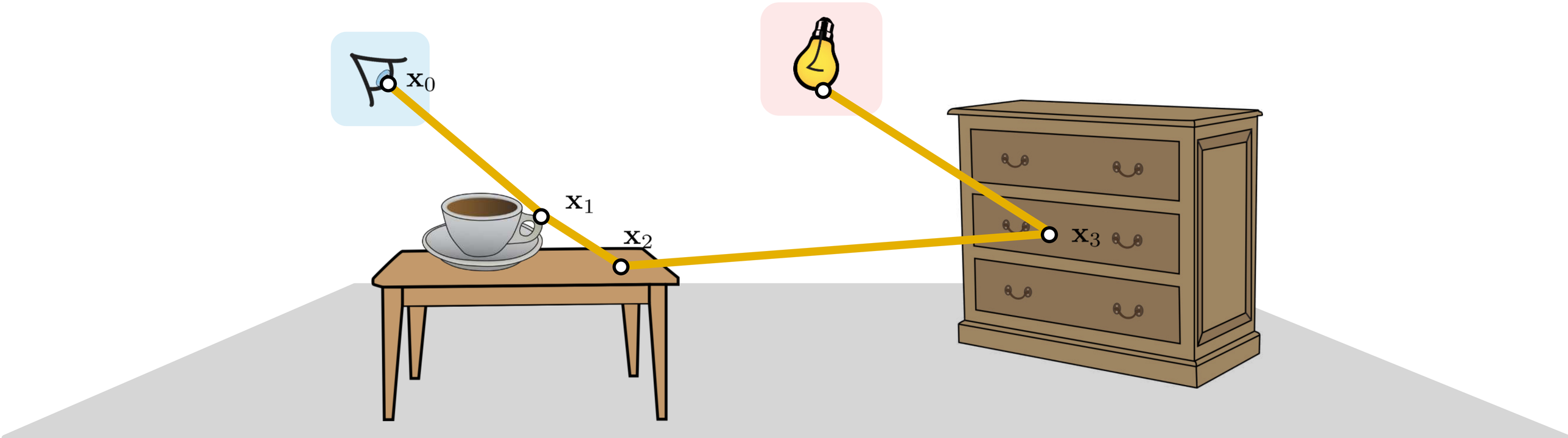
Path integral form of light transport

sensor weight

source weight

image $I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$ light path

space of all light paths



Path integral form of light transport

sensor weight

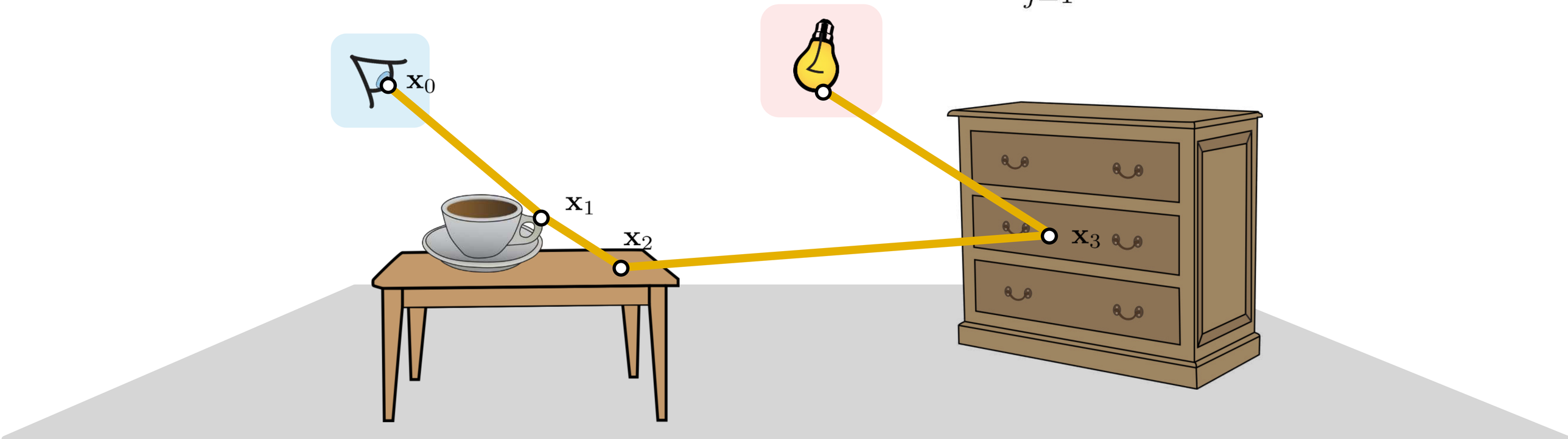
source weight

image $I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$ light path

space of all light paths

path throughput

$$T(\bar{\mathbf{x}}) = G(\mathbf{x}_0, \mathbf{x}_1) \prod_{j=1}^{k-1} f(\mathbf{x}_j, \mathbf{x}_{j+1}, \mathbf{x}_{j-1}) G(\mathbf{x}_j, \mathbf{x}_{j+1})$$



Path integral form of light transport

sensor weight

source weight

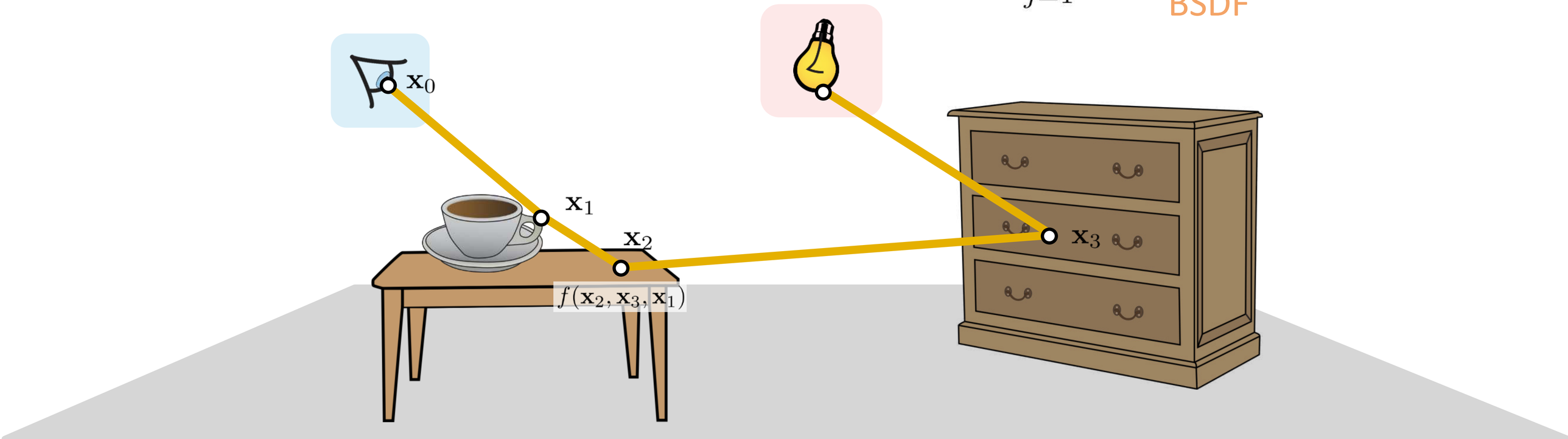
image $I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$ light path

space of all light paths

path throughput

$$T(\bar{\mathbf{x}}) = G(\mathbf{x}_0, \mathbf{x}_1) \prod_{j=1}^{k-1} f(\mathbf{x}_j, \mathbf{x}_{j+1}, \mathbf{x}_{j-1}) G(\mathbf{x}_j, \mathbf{x}_{j+1})$$

BSDF



Path integral form of light transport

sensor weight

source weight

image $I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$ light path

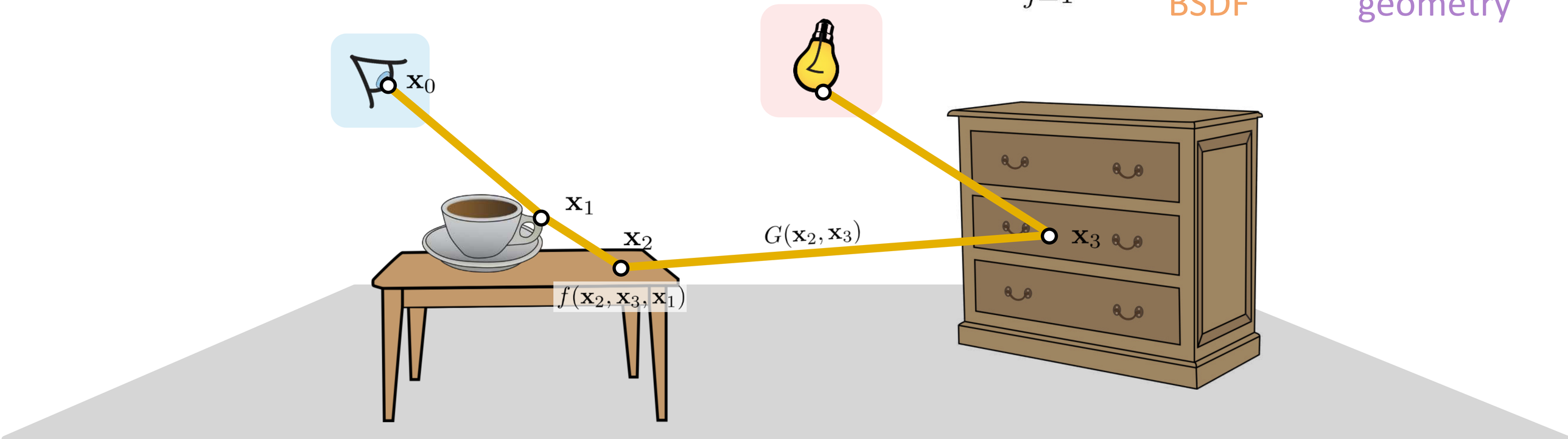
space of all light paths

path throughput

$$T(\bar{\mathbf{x}}) = G(\mathbf{x}_0, \mathbf{x}_1) \prod_{j=1}^{k-1} f(\mathbf{x}_j, \mathbf{x}_{j+1}, \mathbf{x}_{j-1}) G(\mathbf{x}_j, \mathbf{x}_{j+1})$$

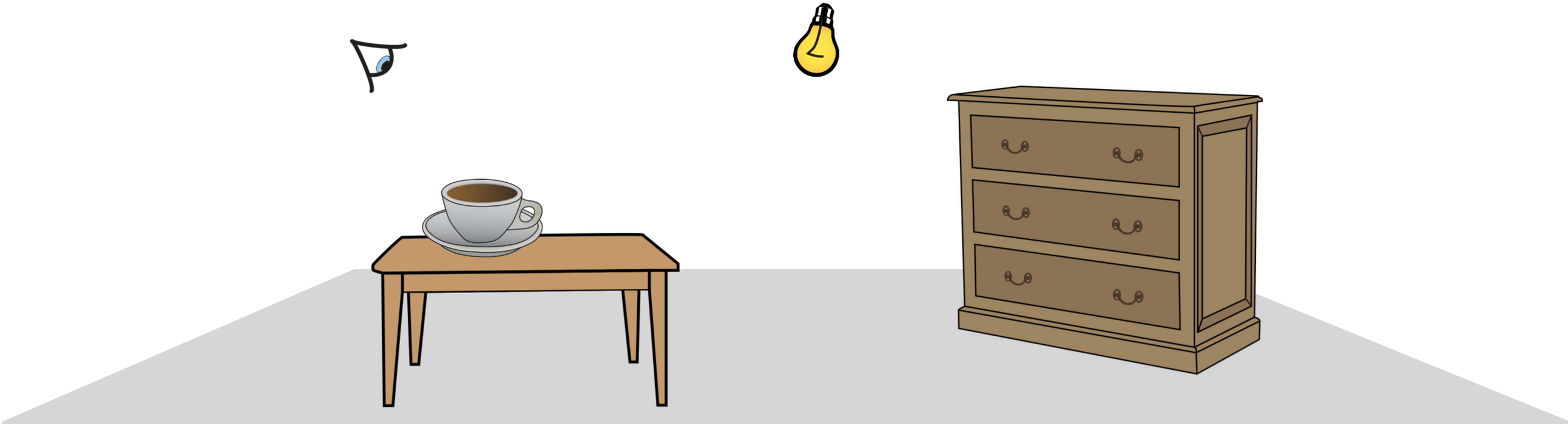
BSDF

geometry



Monte Carlo rendering

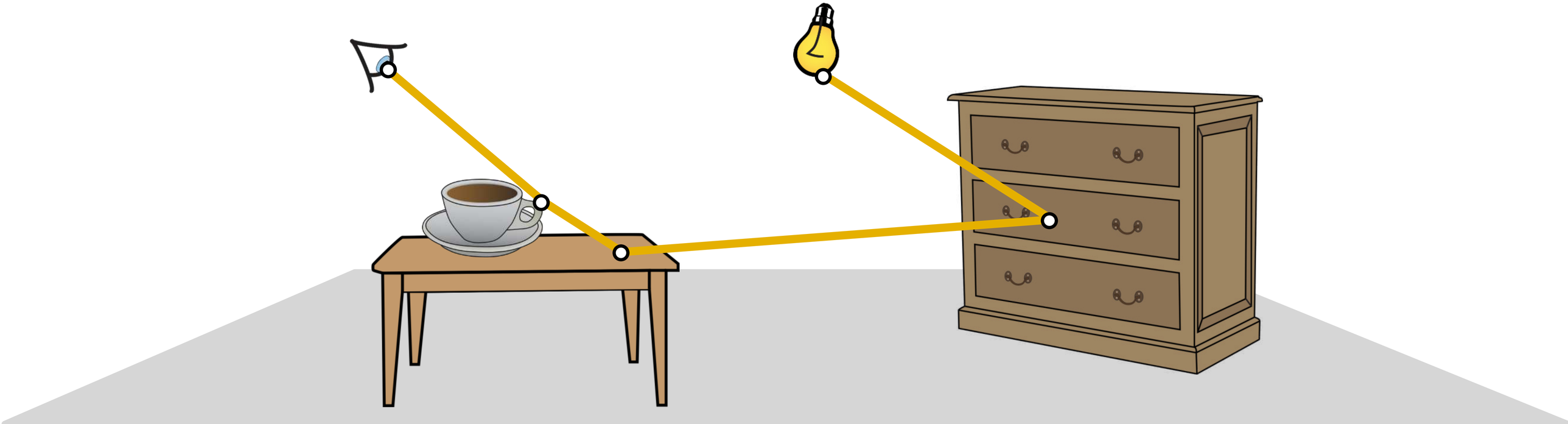
approximate image as $I \approx \frac{1}{N}\sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$



Monte Carlo rendering

approximate image as $I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

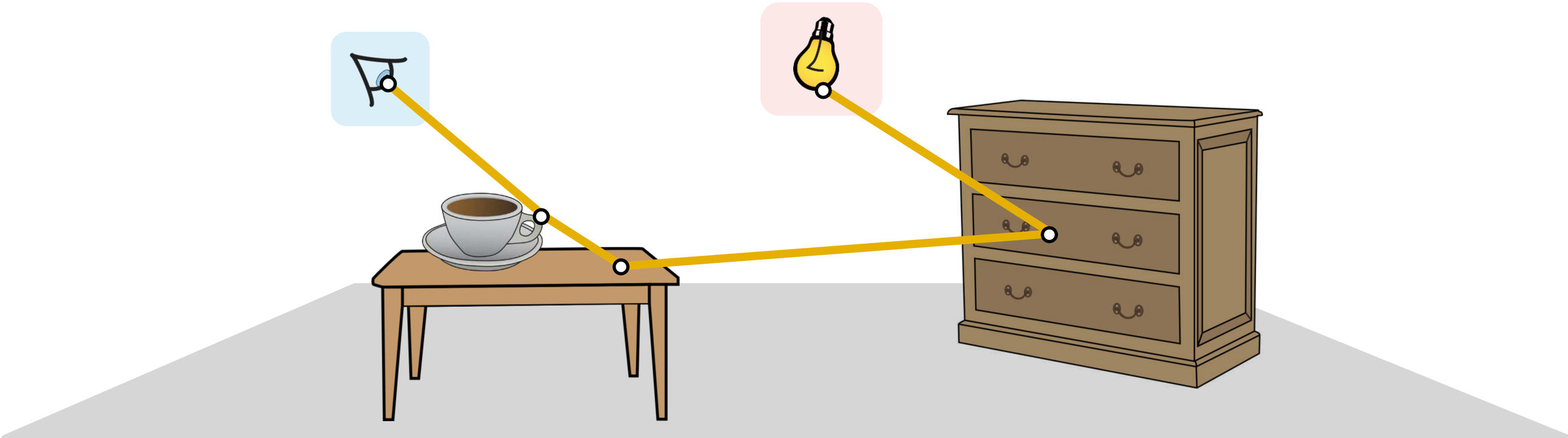


Monte Carlo rendering

sensor weight source weight path throughput

approximate image as $I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths



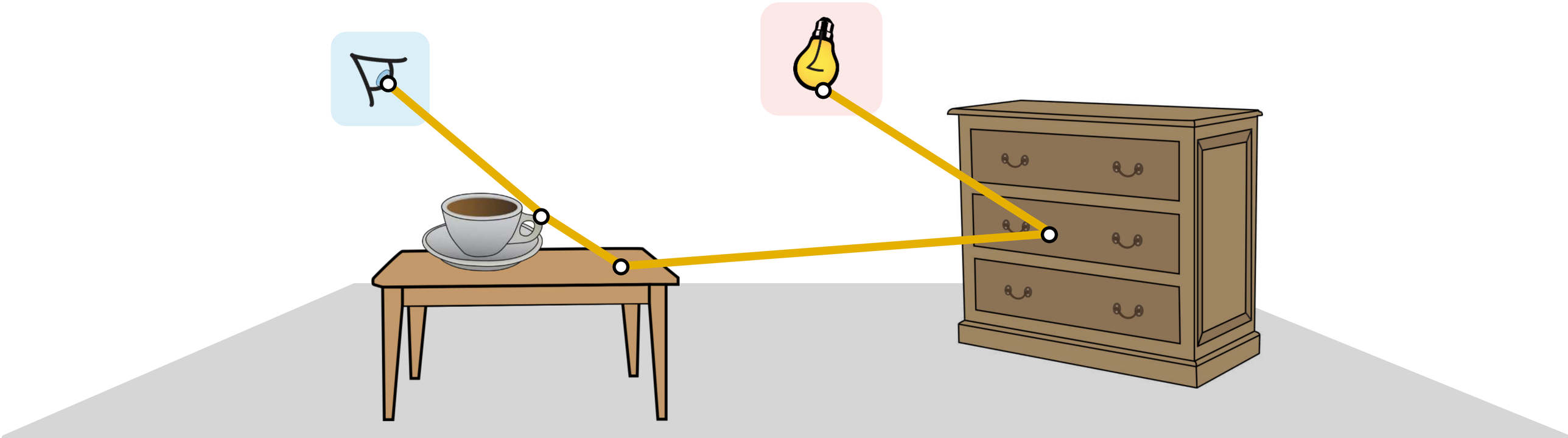
Monte Carlo rendering

sensor weight source weight path throughput

approximate image as $I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path



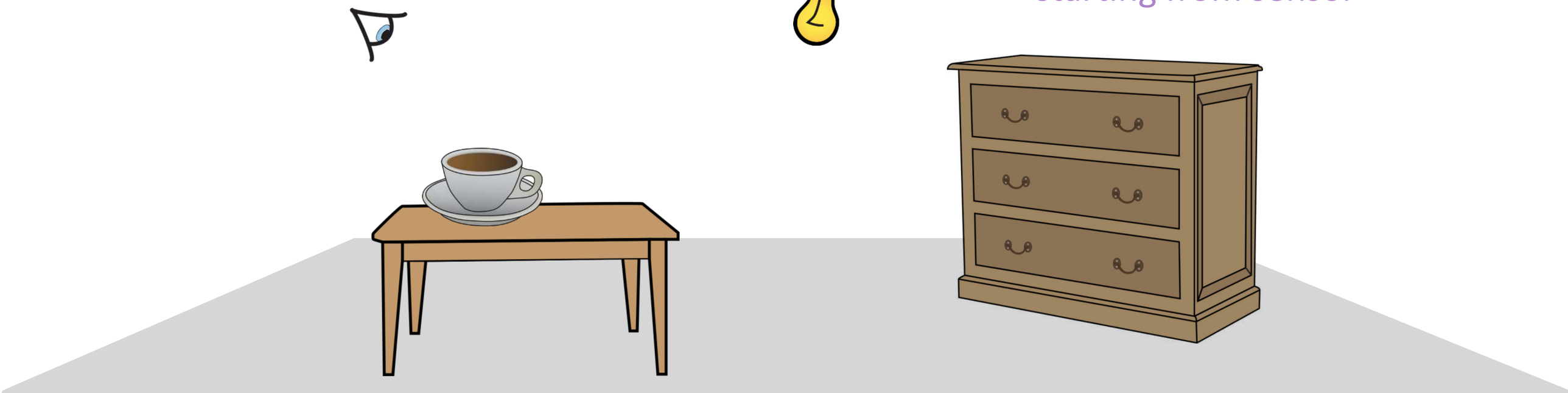
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Path tracing: sample path starting from sensor



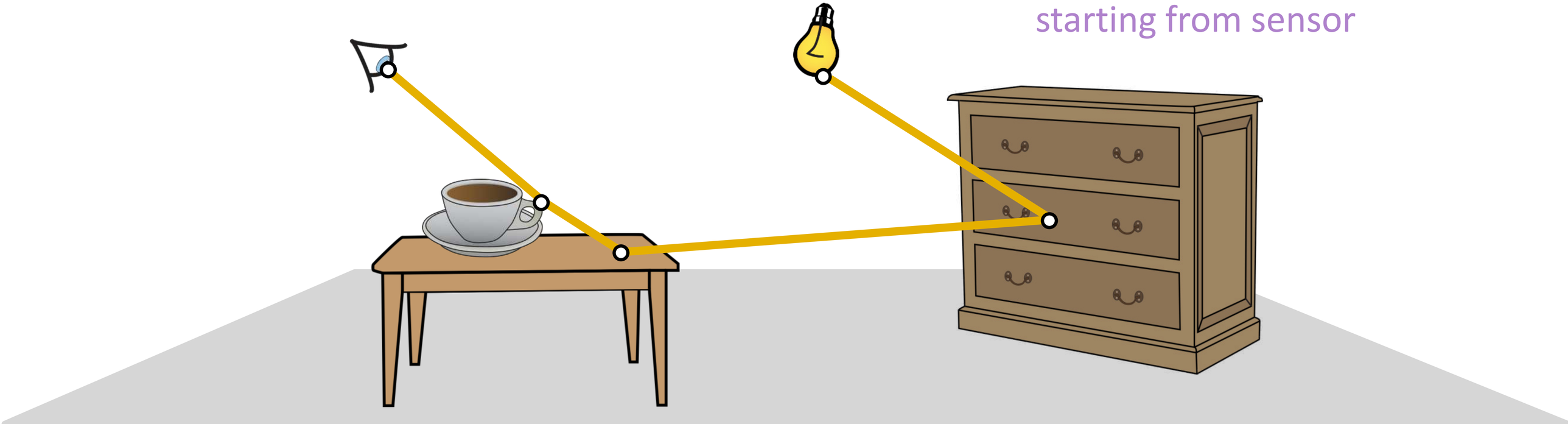
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Path tracing: sample path starting from sensor



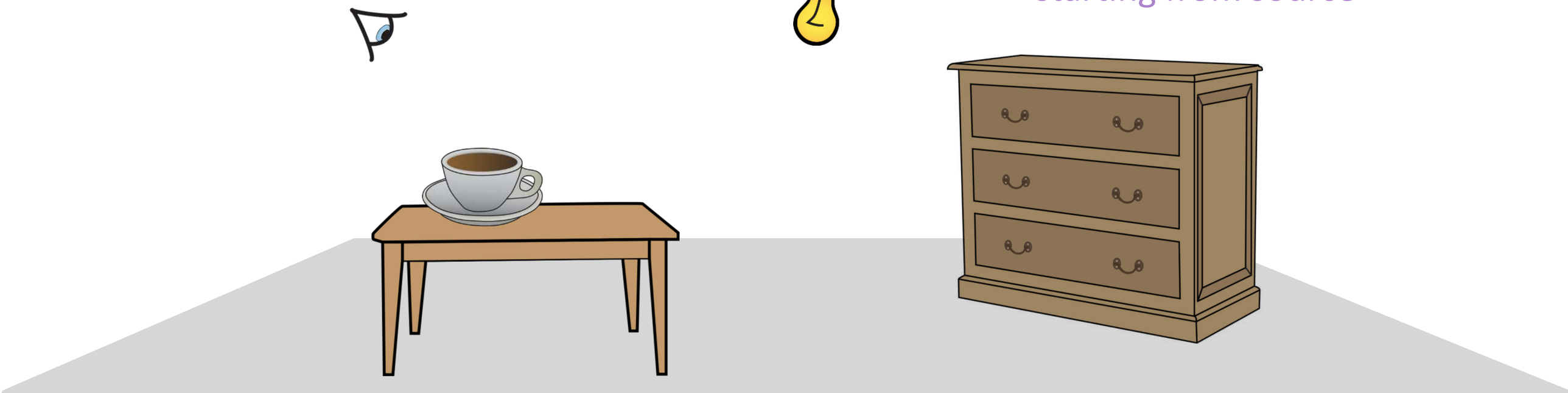
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Light tracing: sample path starting from source



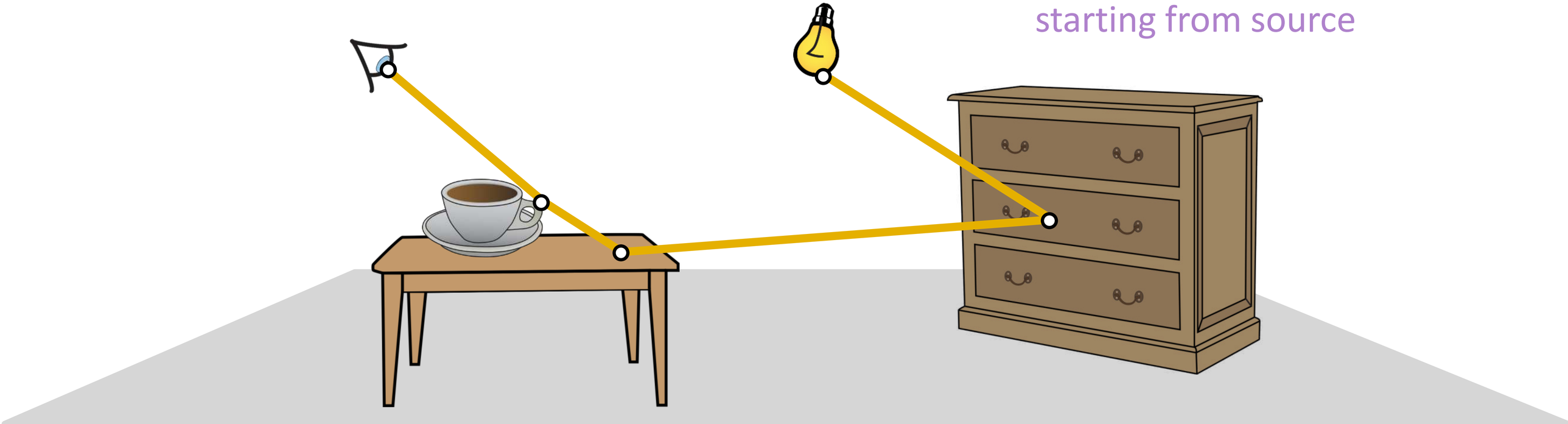
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Light tracing: sample path starting from source



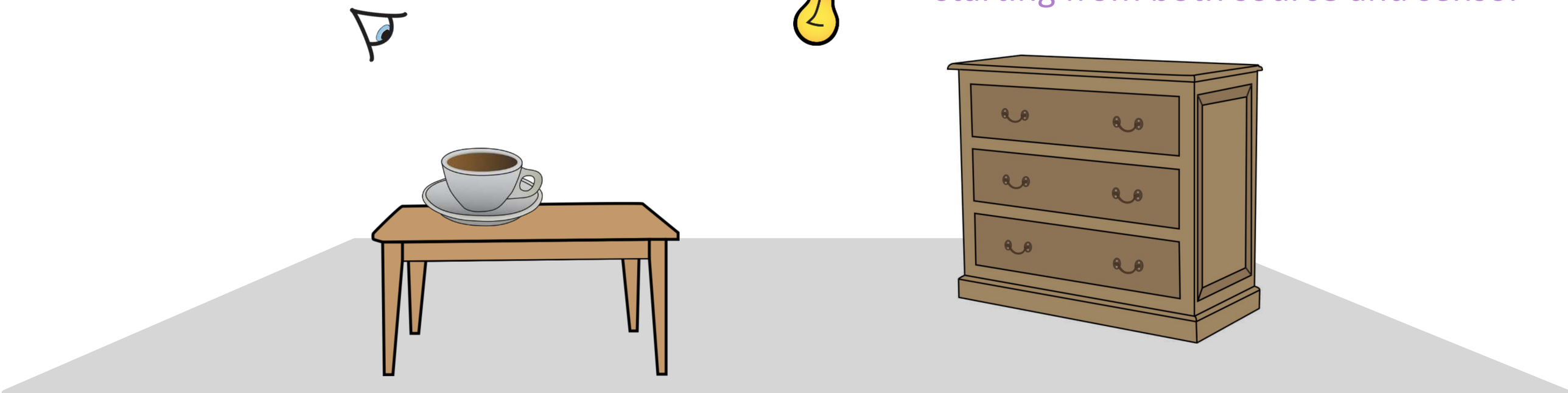
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Bidirectional path tracing: sample path starting from both source and sensor



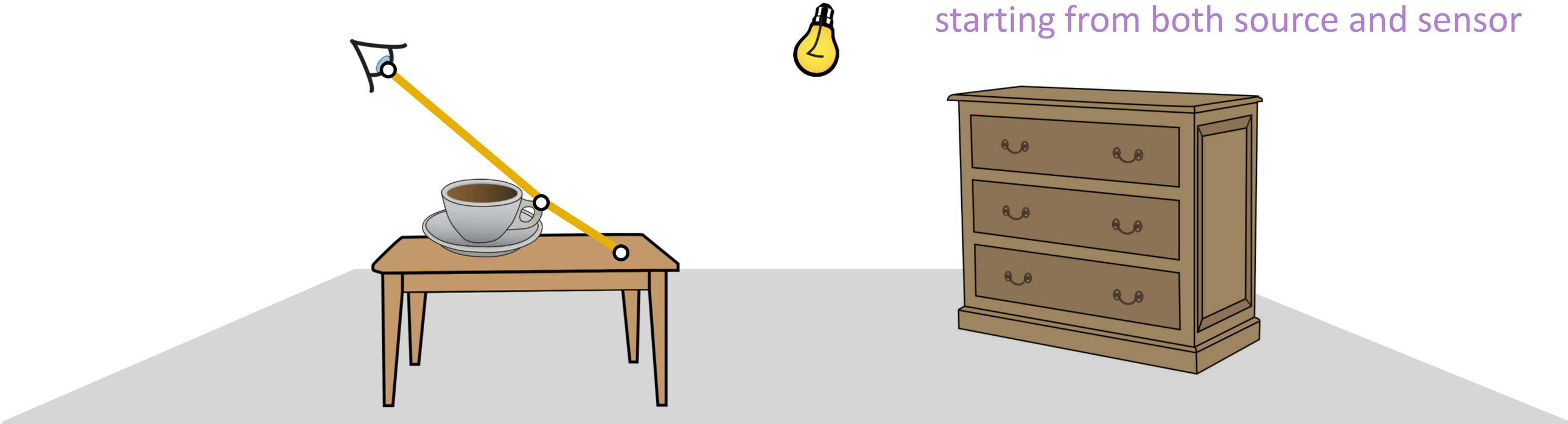
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Bidirectional path tracing: sample path starting from both source and sensor



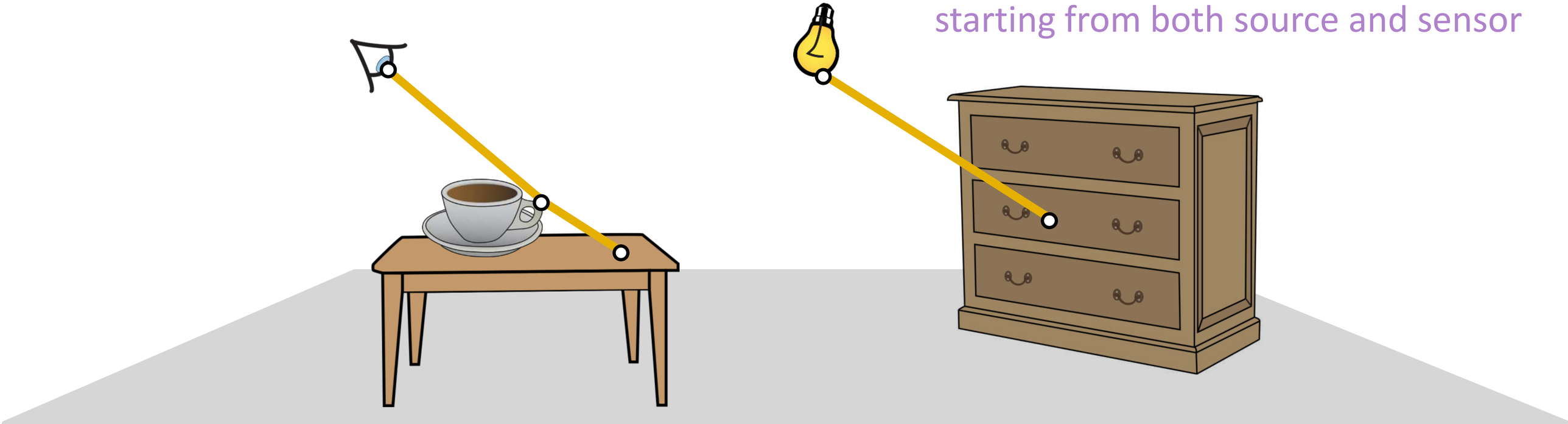
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Bidirectional path tracing: sample path starting from both source and sensor



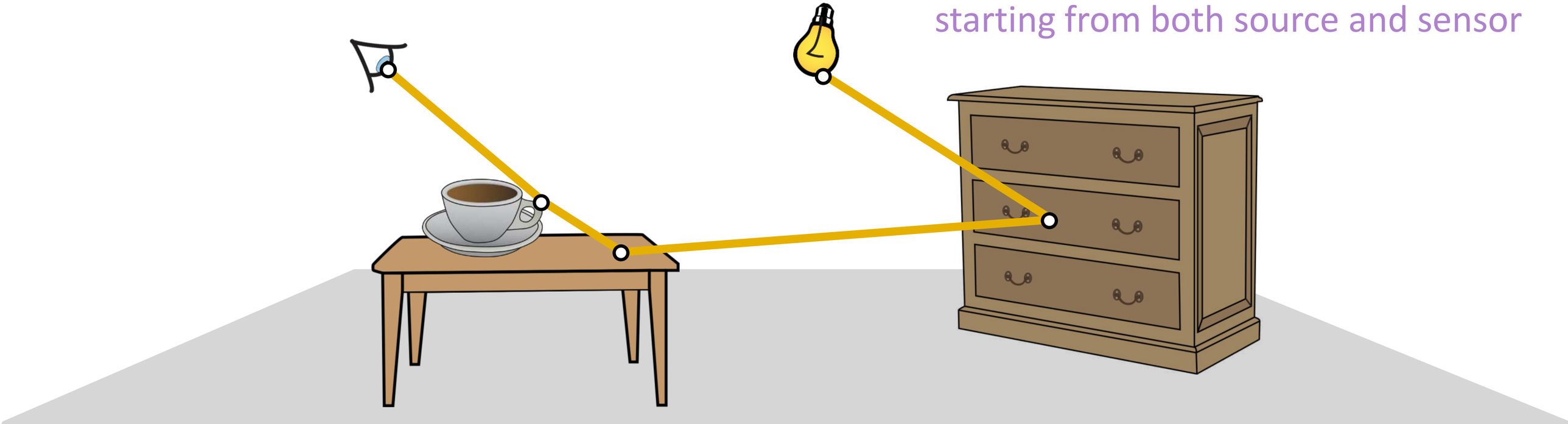
Monte Carlo rendering

approximate image as $I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{W_e(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}) L_e(\mathbf{x}_{i,k}, \mathbf{x}_{i,k-1}) T(\bar{\mathbf{x}}_i)}{p(\bar{\mathbf{x}}_i)}$

sum over *randomly* sampled paths

PDF of random path

Bidirectional path tracing: sample path starting from both source and sensor

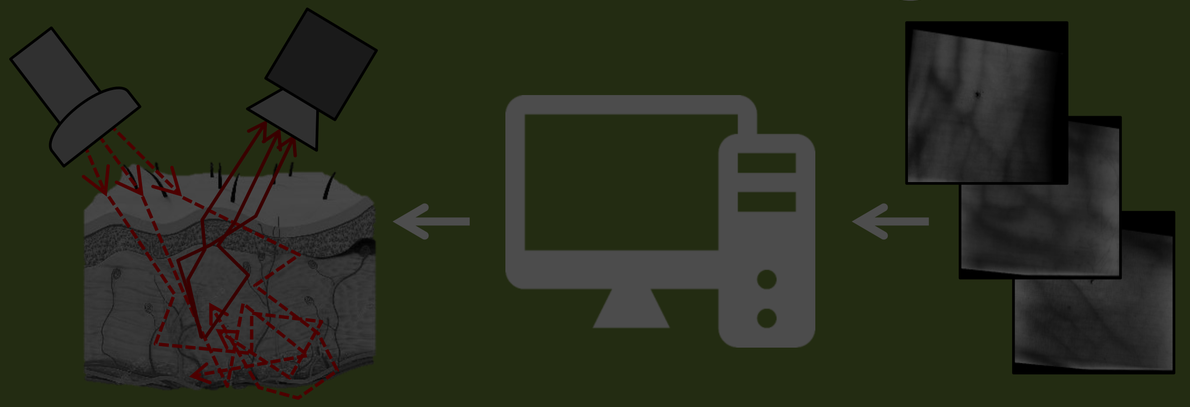


Physics-based rendering and its applications to computational imaging

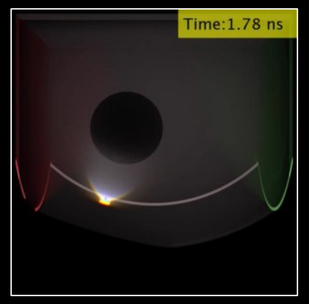
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

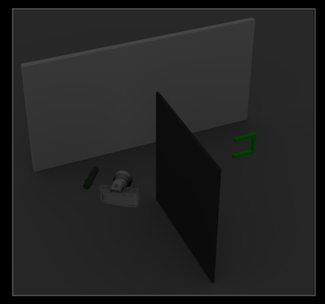
inverse rendering



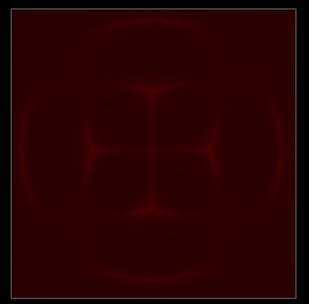
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



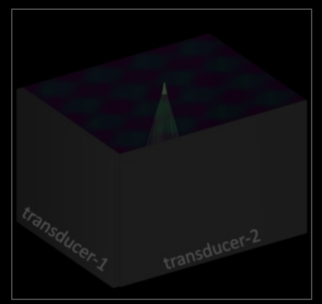
time-of-flight imaging



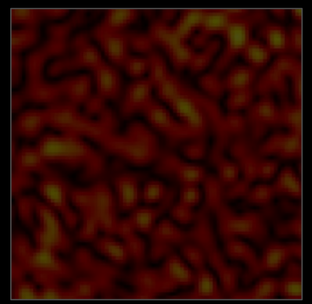
non-line-of-sight imaging



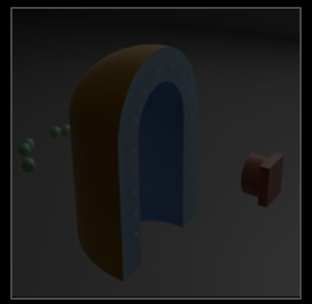
acousto-optic lensing



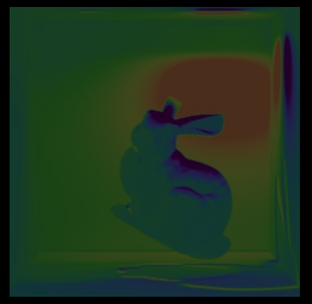
ultrafast light scanning



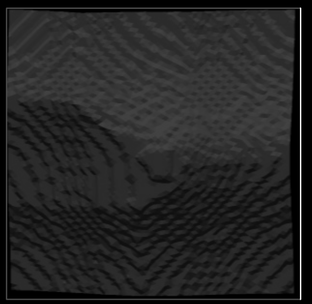
speckle imaging



tactile sensor design



differentiable renderer

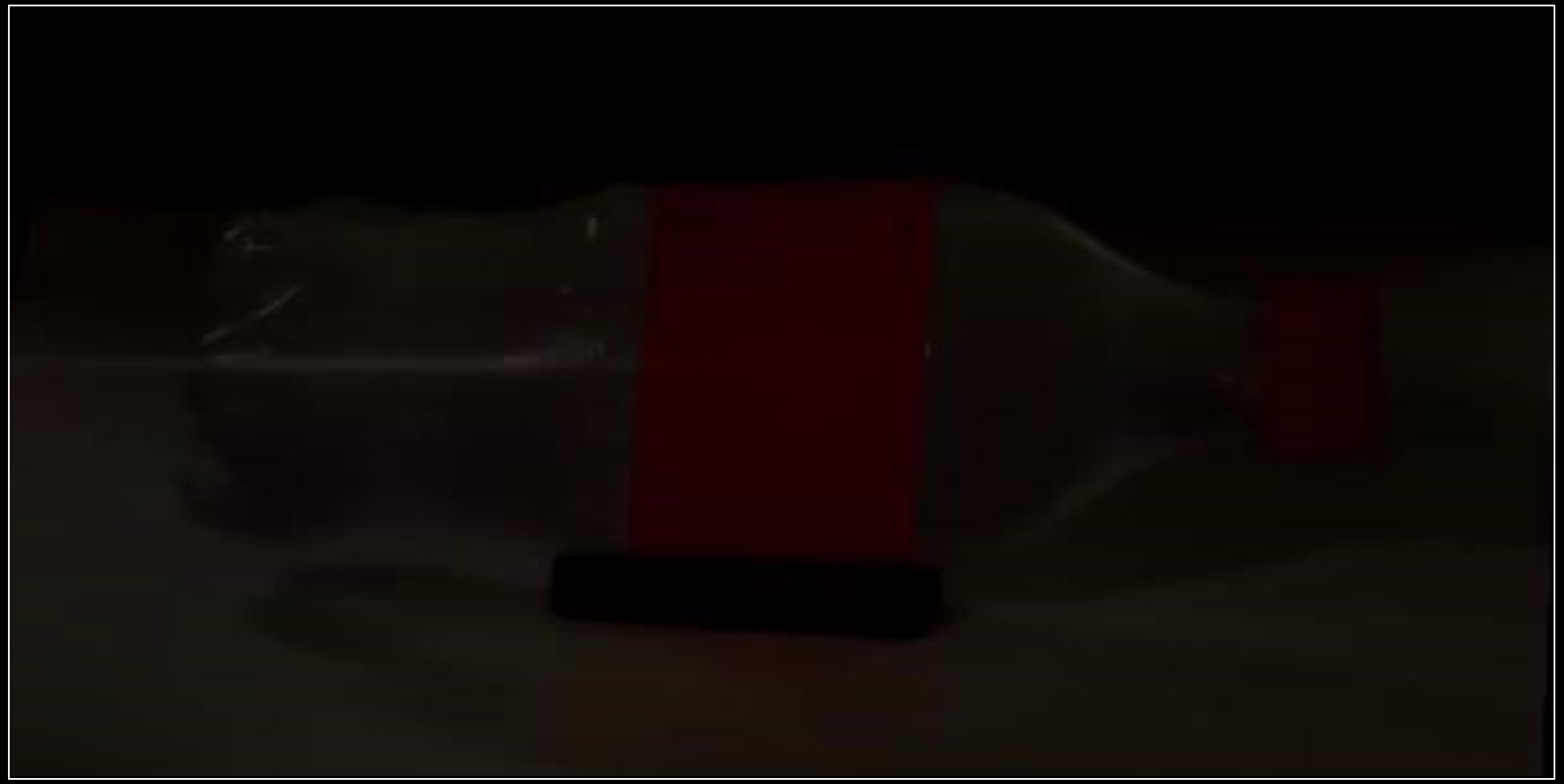


inverse problems

Time-of-flight cameras

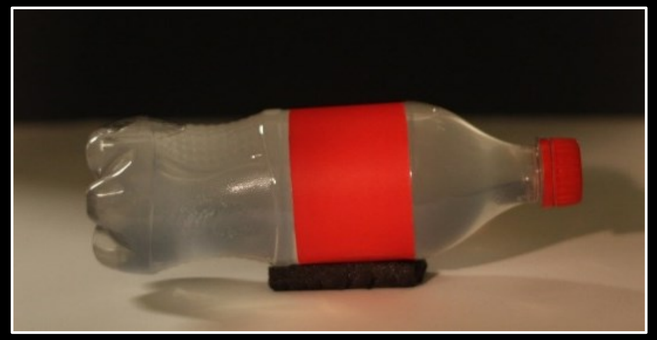
[Velten et al. ToG, 2013]

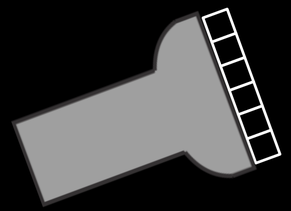
Time-of-flight cameras



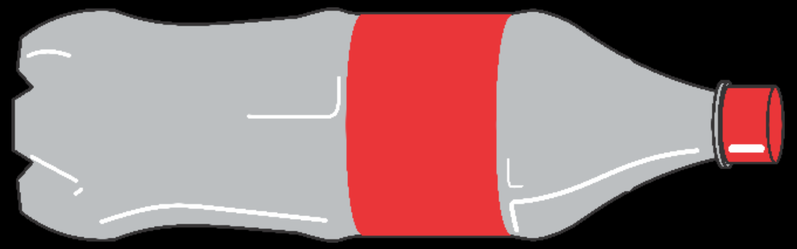
[Velten et al. ToG, 2013]

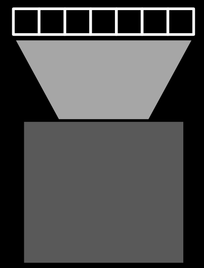
Time-of-flight cameras





light
source

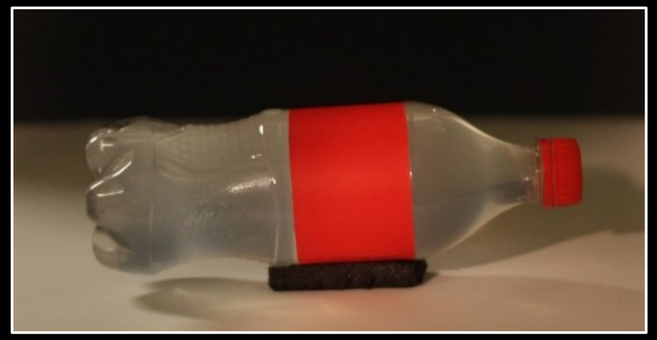


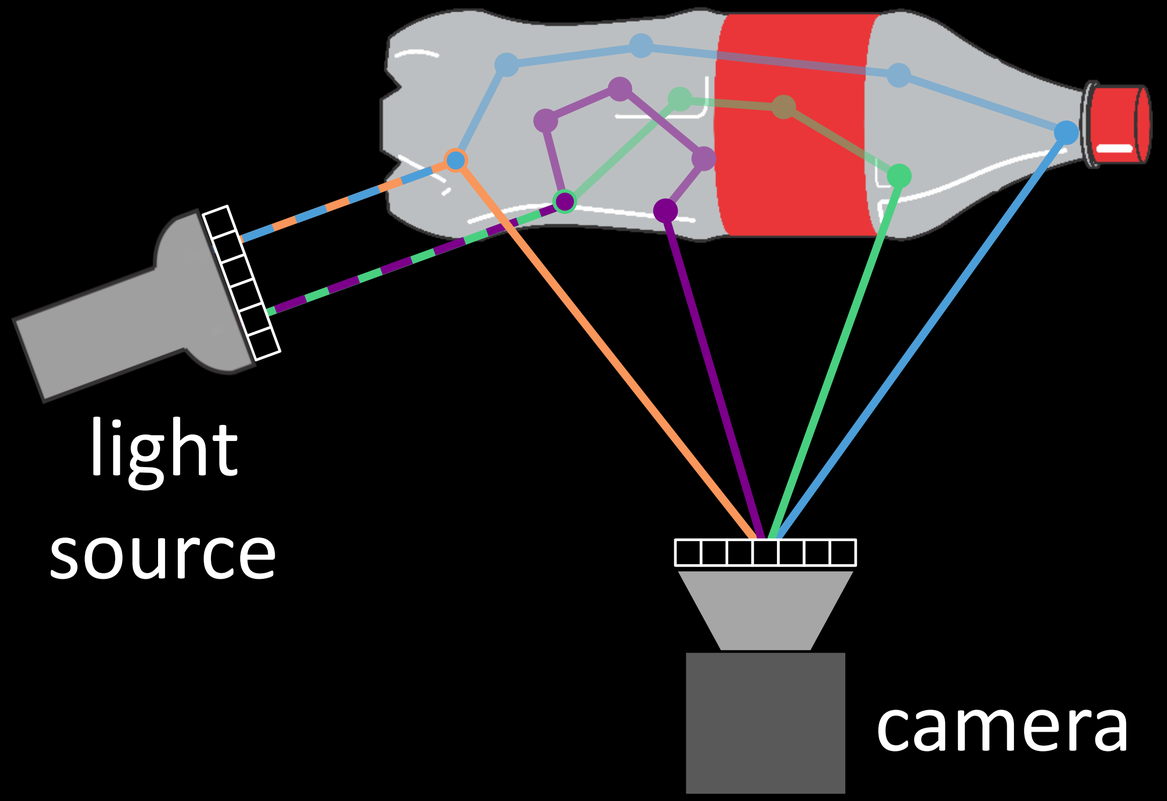


camera

[Velten et al. ToG, 2013]

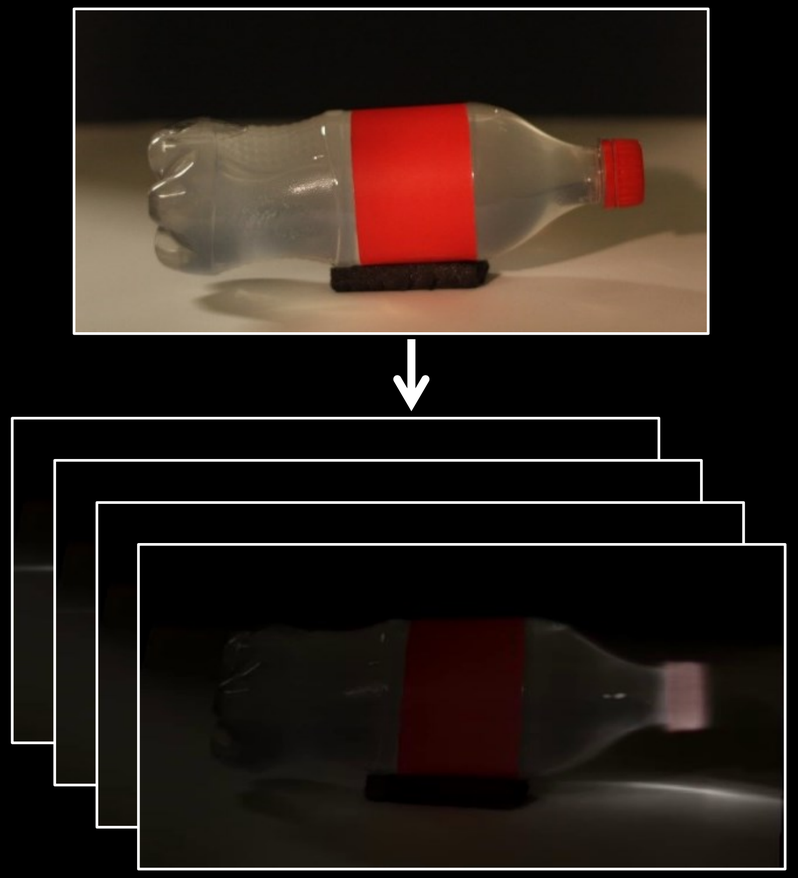
Time-of-flight cameras

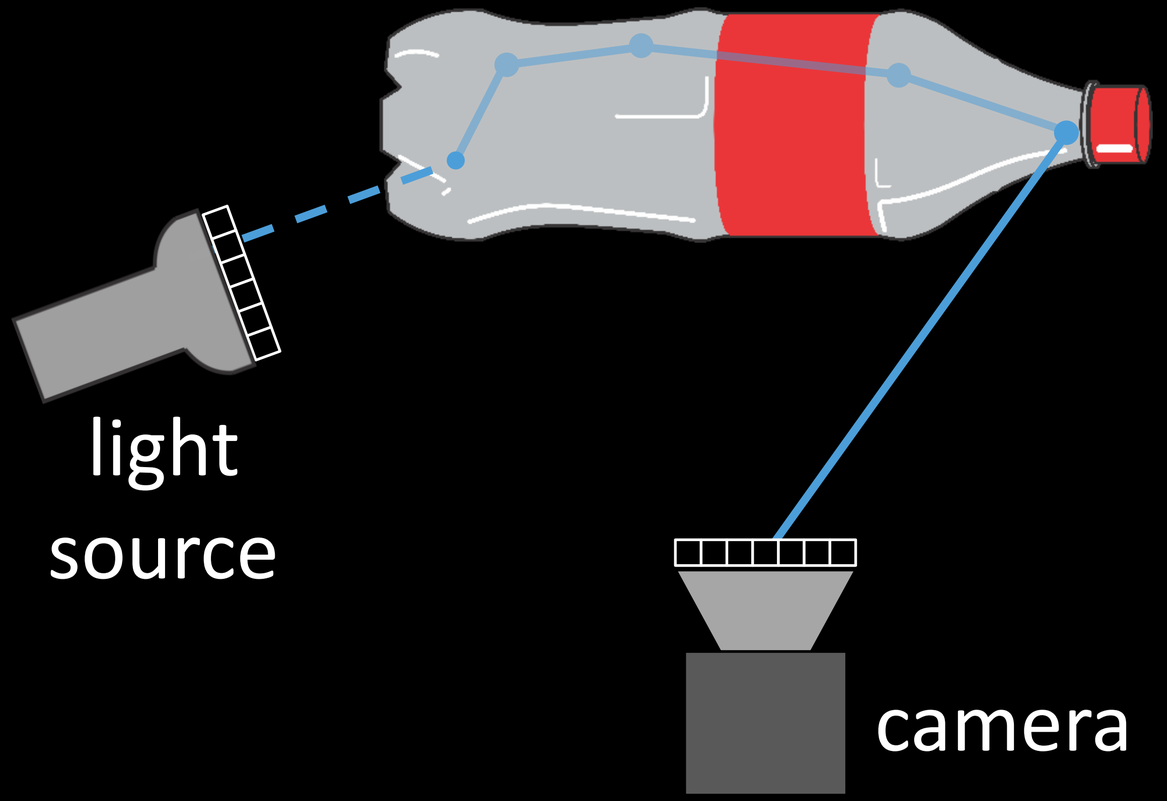




[Velten et al. ToG, 2013]

Time-of-flight cameras





[Velten et al. ToG, 2013]

Time-of-flight cameras

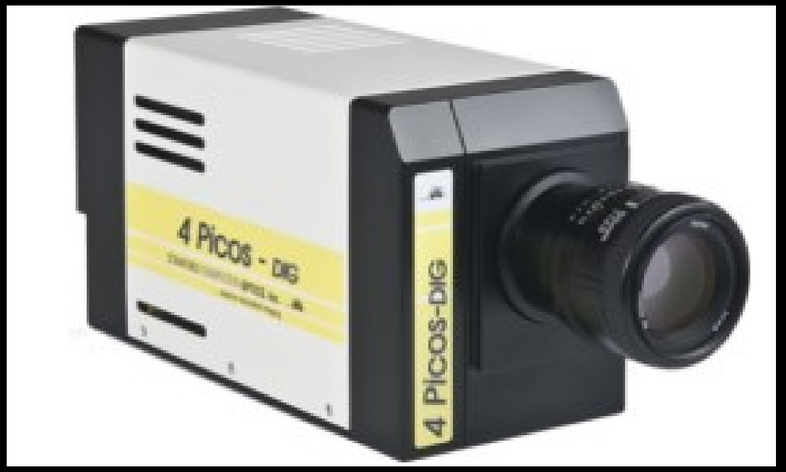
SwissRanger SR4000

Hamamatsu streak

Brightway VISDOM

Microsoft Kinect

MPD SPAD



Stanford ICCD

Time-of-flight cameras

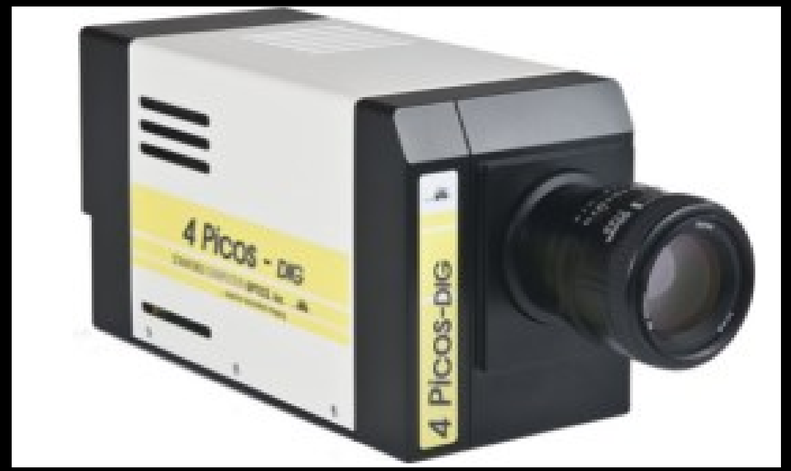
SwissRanger SR4000

Hamamatsu streak

Brightway VISDOM

Microsoft Kinect

MPD SPAD



Stanford ICCD

continuous-wave

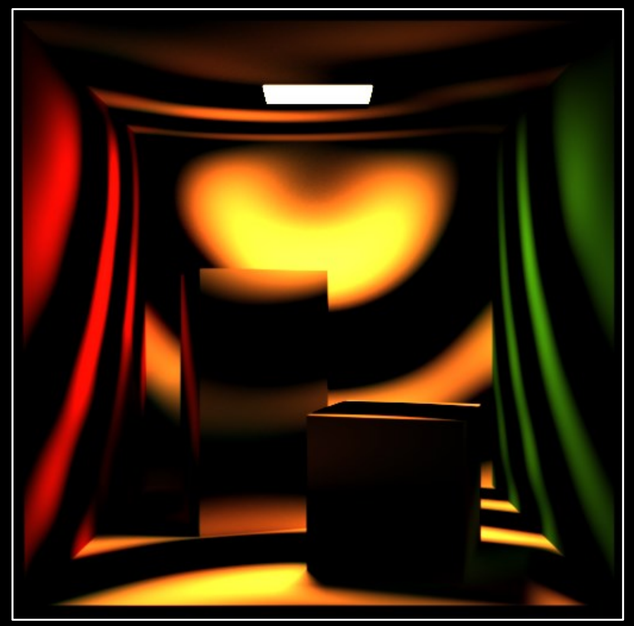
transient

time-gated

Time-of-flight cameras

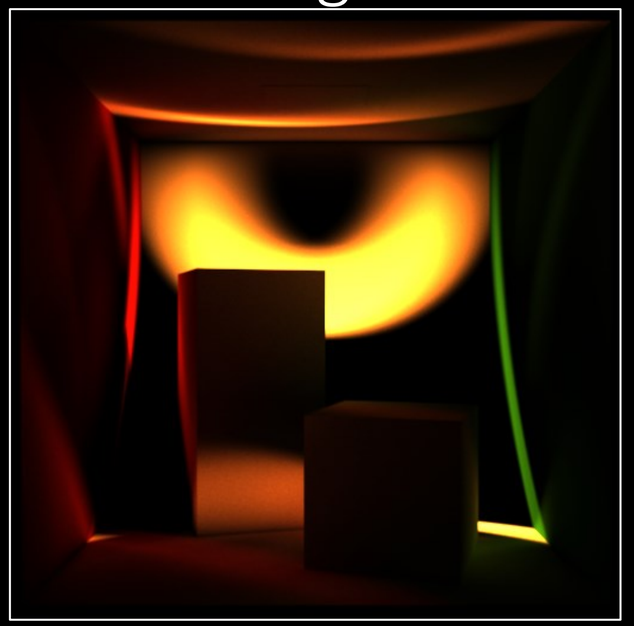
intensity

continuous wave



transient

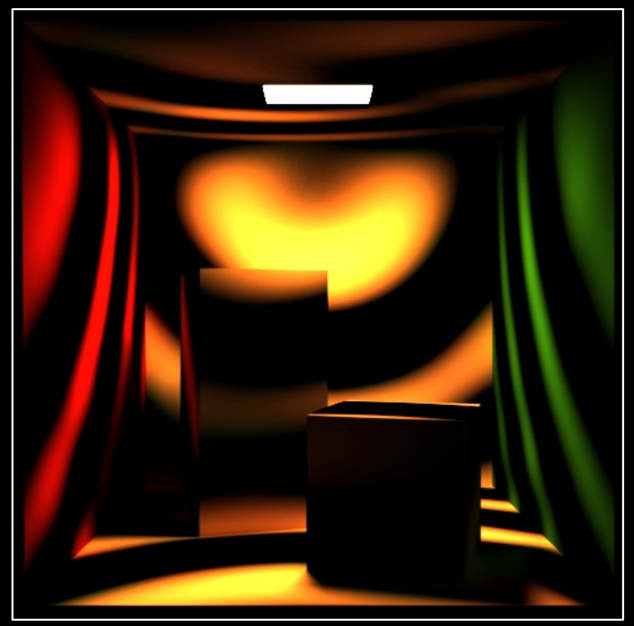
time-gated



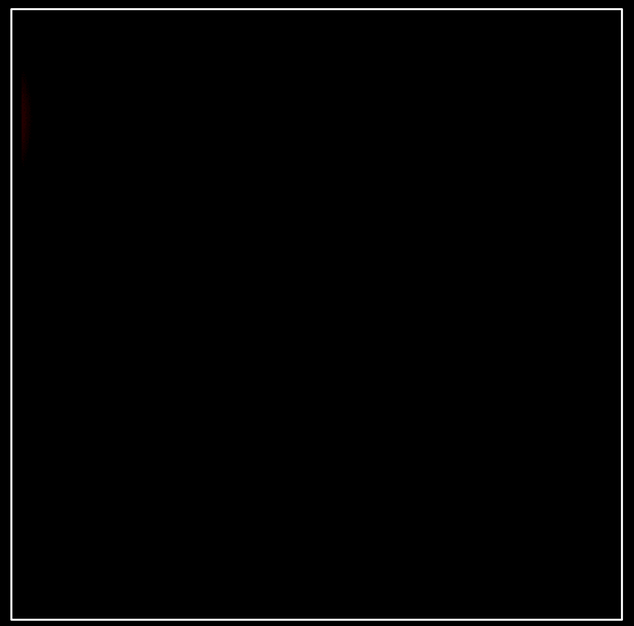
Time-of-flight cameras

intensity

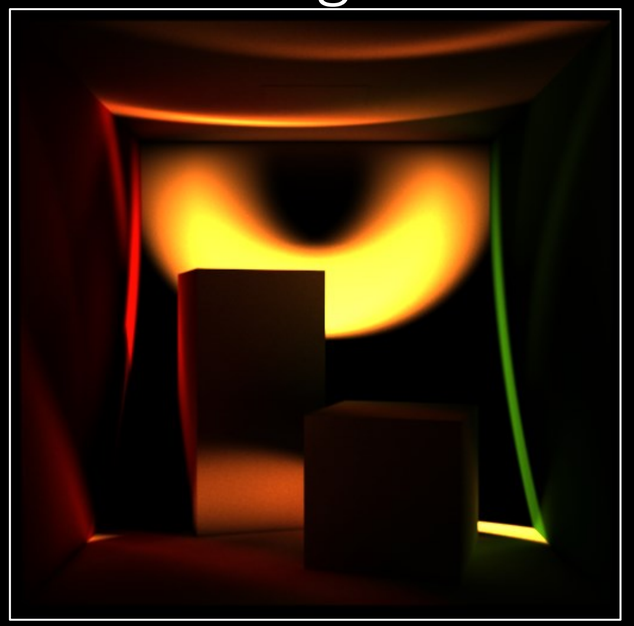
continuous wave



transient



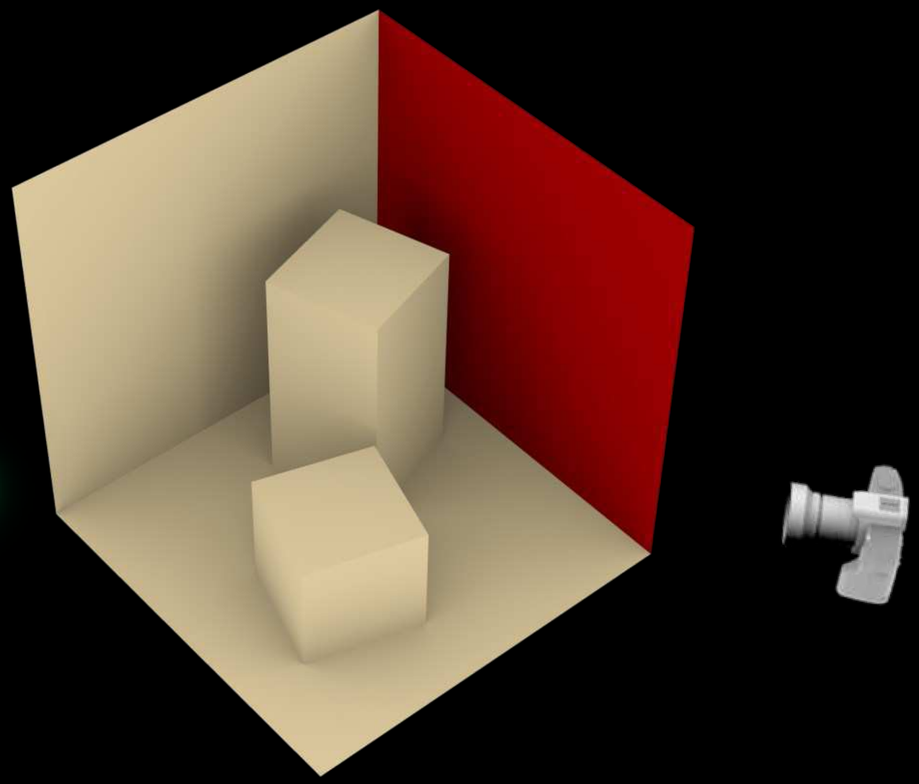
time-gated



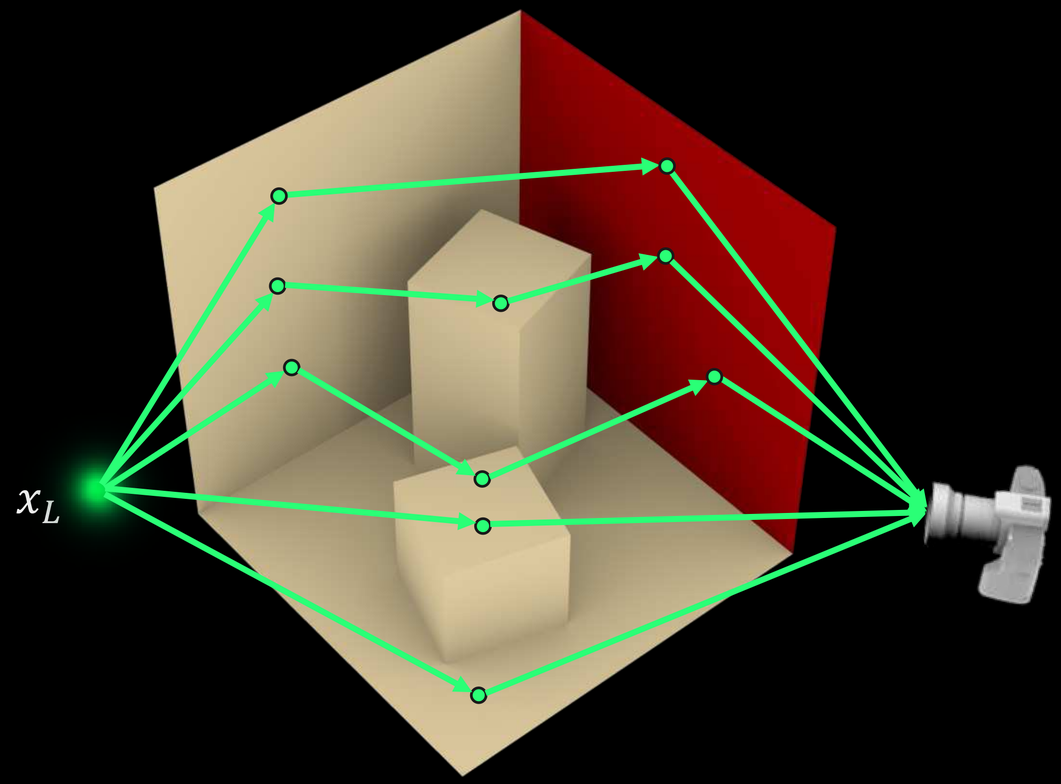
Rendering time-of-flight cameras: Path space integral

$$\text{image} = \int_{\text{light paths}} f(\text{path})$$

path contribution, depends on scene properties, light source, and sensor



Rendering time-of-flight cameras: Path space integral



$$\text{image} = \int_{\text{light paths}} f(\text{path})$$

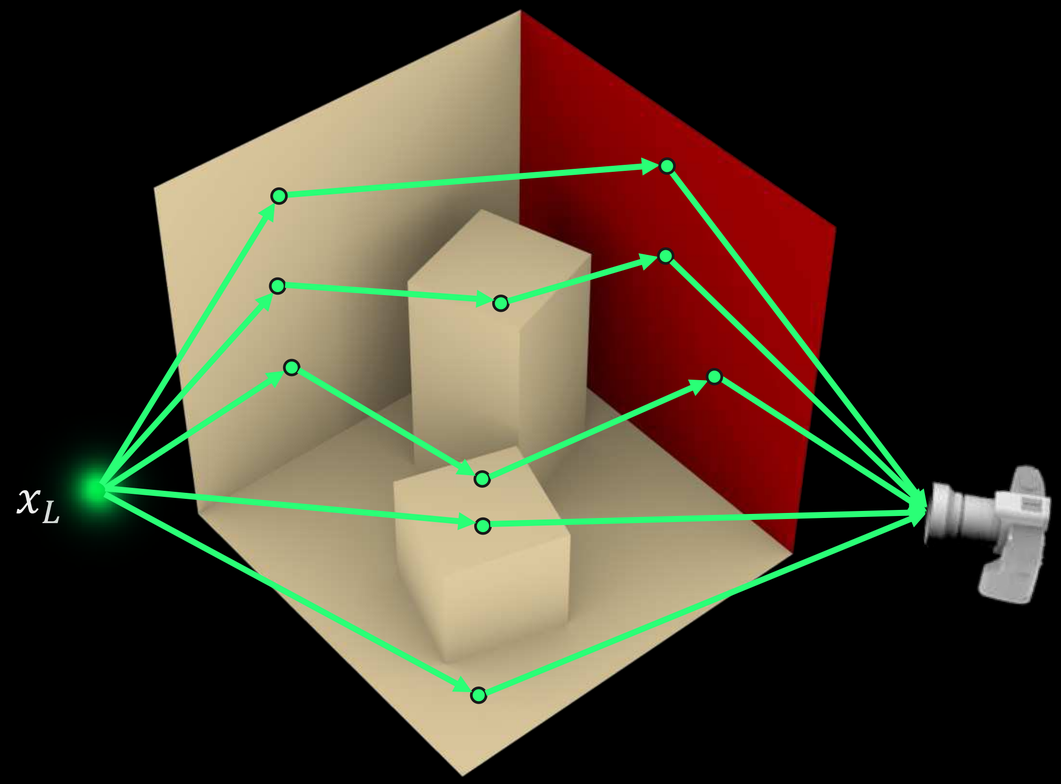
path contribution, depends on scene properties, light source, and sensor

Monte Carlo rendering:

- randomly sample paths: path_1, path_2, …, path_N
- approximate image as:

$$\text{image} \approx \sum_n \frac{f(\text{path}_n)}{\text{prob}(\text{path}_n)}$$

Rendering time-of-flight cameras: Path space integral



$$\text{image} = \int_{\text{light paths}} f(\text{path})\, W(|\text{path}|)$$

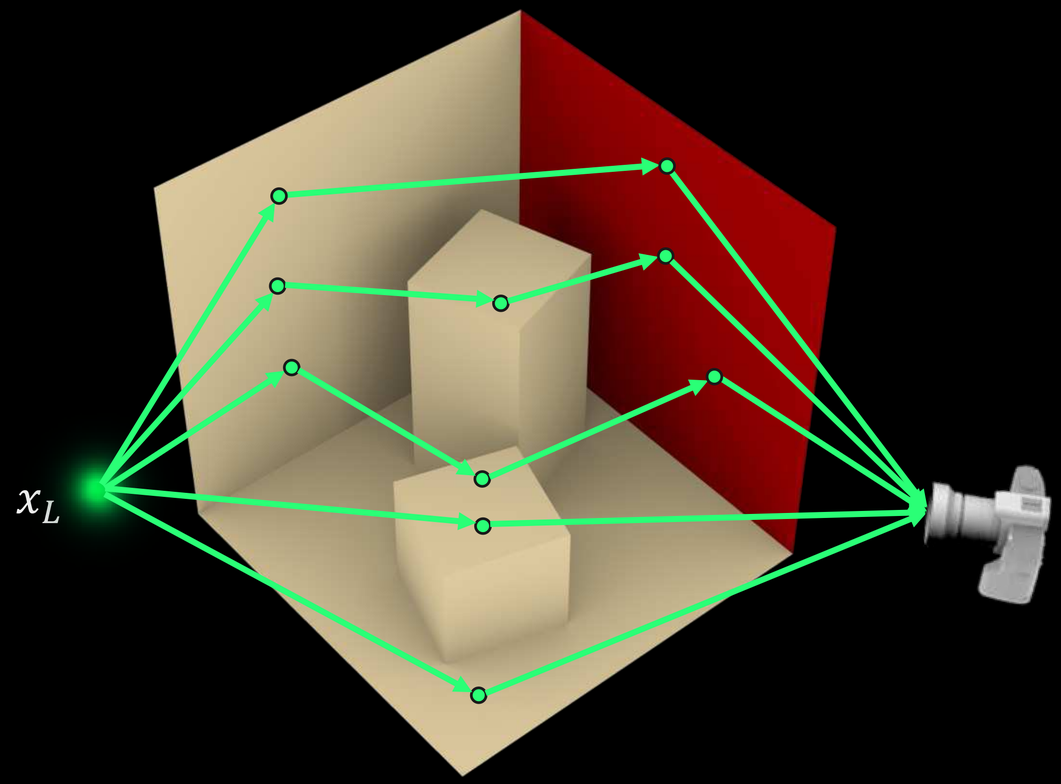
path length weight depends only on sensor

Monte Carlo rendering:

- randomly sample paths: path_1, path_2, …, path_N
- approximate image as:

$$\text{image} \approx \sum_n \frac{f(\text{path}_n)}{\text{prob}(\text{path}_n)}$$

Rendering time-of-flight cameras: Path space integral



$$\text{image} = \int_{\substack{\text{light} \\ \text{paths}}} \mathbf{f(path)}\, W(|\text{path}|)$$

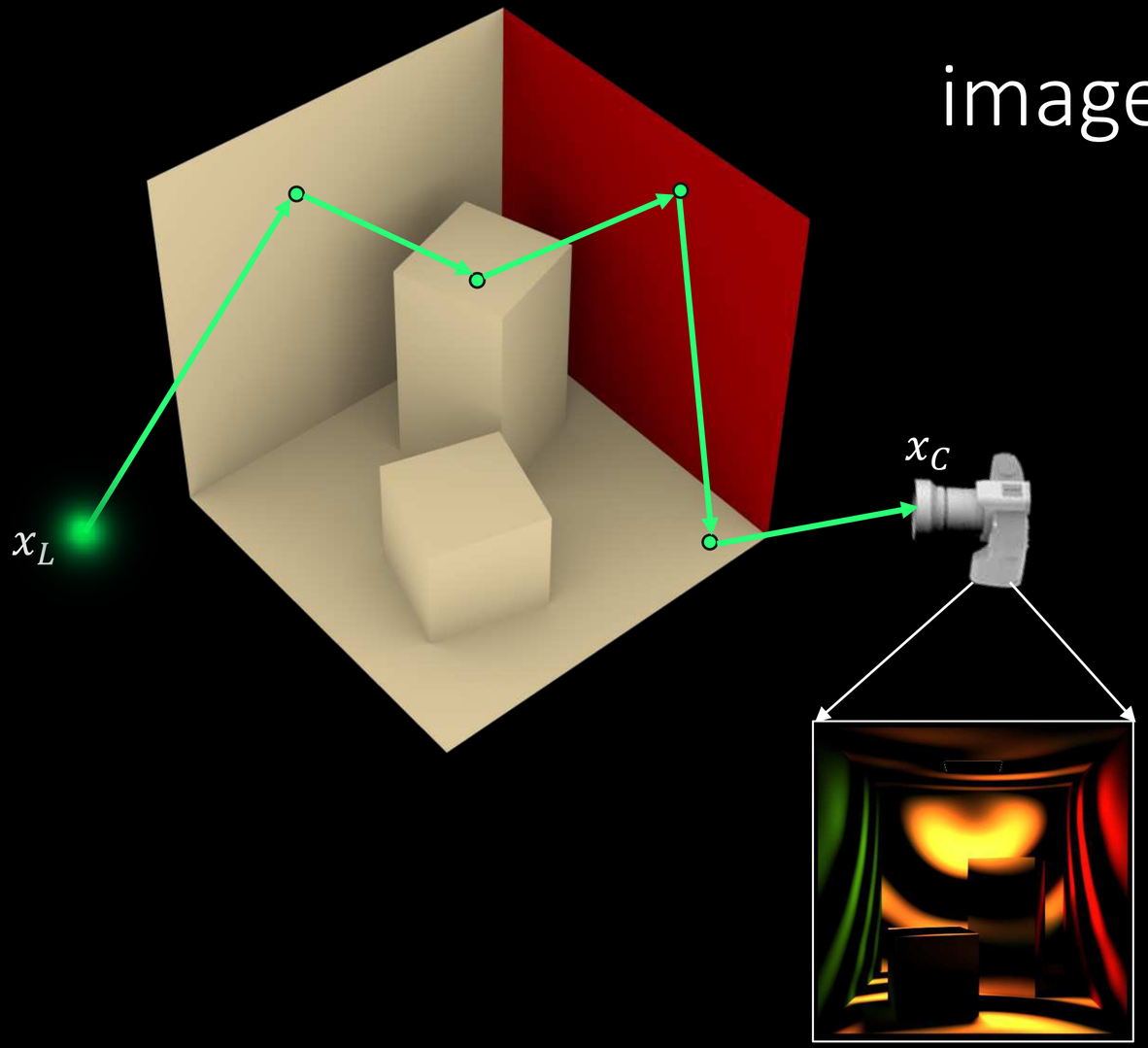
path length weight
depends only on sensor

Monte Carlo rendering:

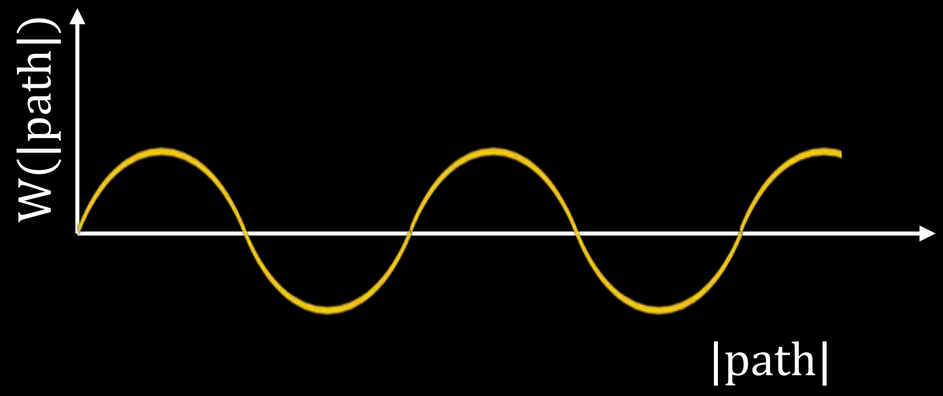
- randomly sample paths: path_1, path_2, …, path_N
- approximate image as:

$$\text{time-of-flight image} \approx \sum_n \frac{f(\text{path}_n)}{\text{prob}(\text{path}_n)} W(|\text{path}_n|)$$

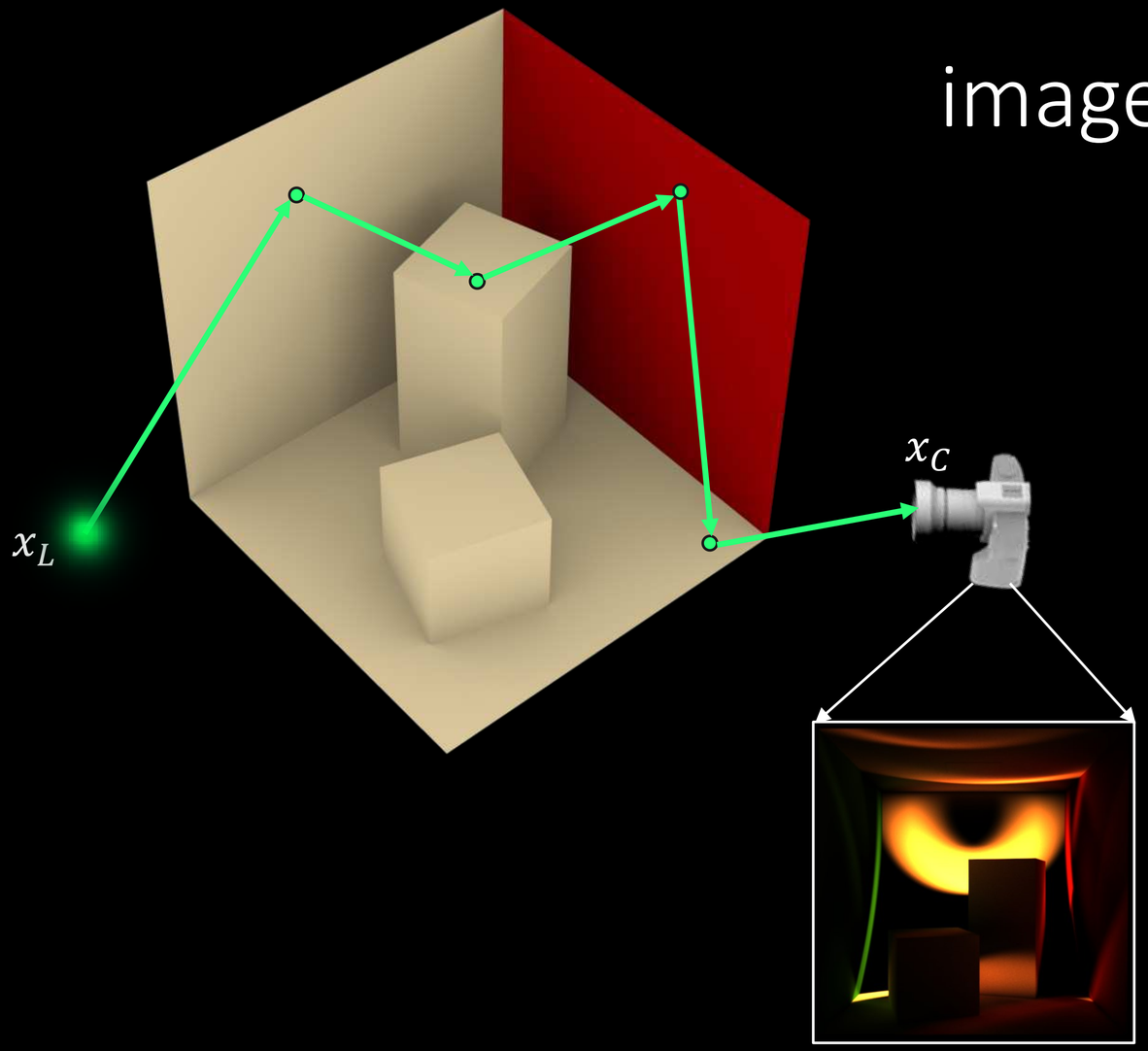
Rendering continuous wave time-of-flight camera



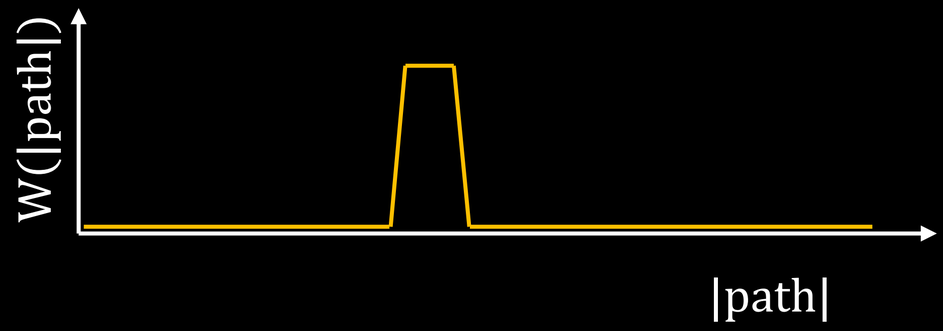
$$\text{image} = \int_{\text{light paths}} f(\text{path})\,W(|\text{path}|)$$



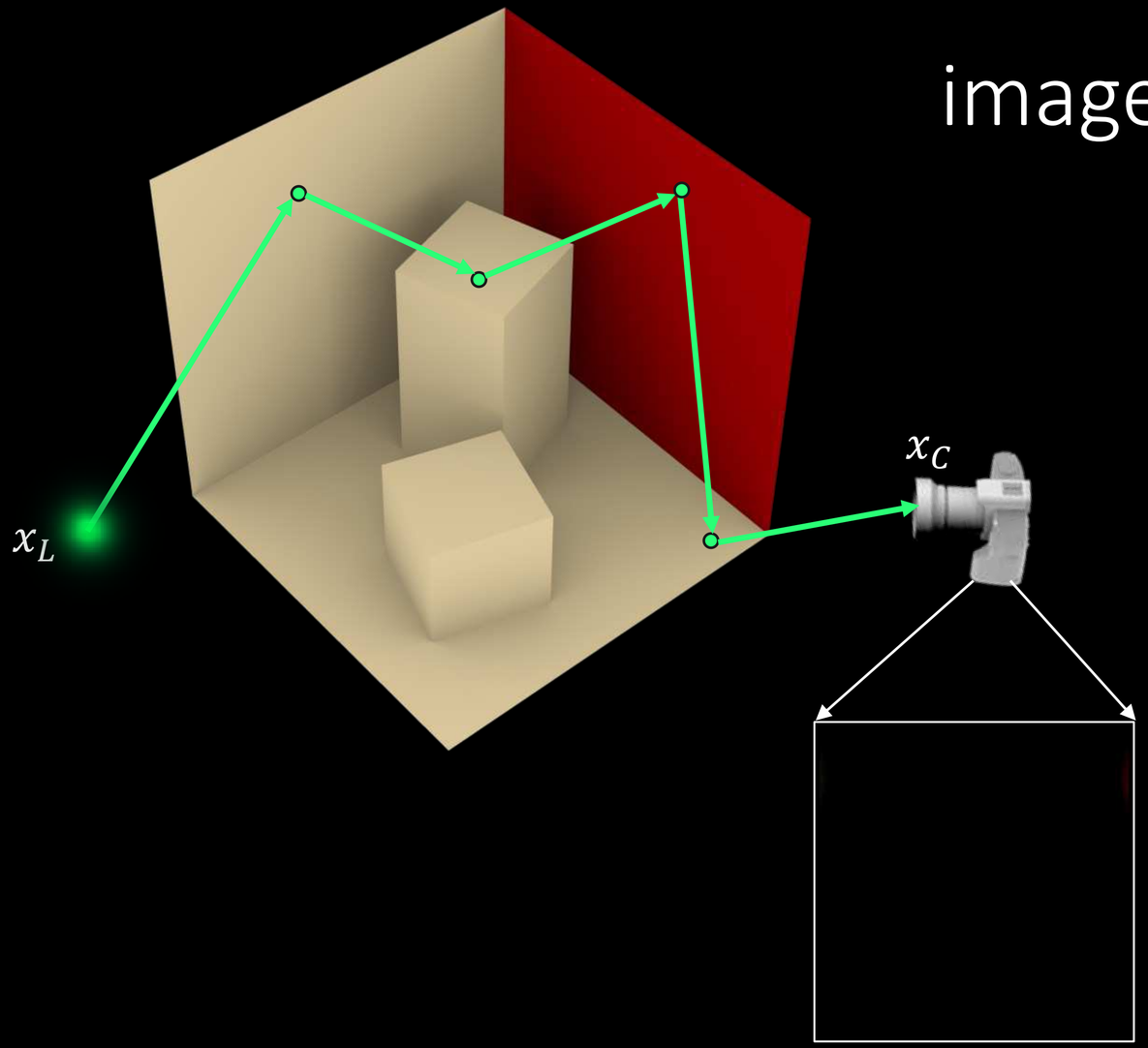
Classification: time-gated camera



$$\text{image} = \int_{\text{light paths}} f(\text{path})\,W(|\text{path}|)$$



Rendering transient camera



$$\text{image} = \int_{\text{light paths}} f(\text{path})\,W(|\text{path}|)$$

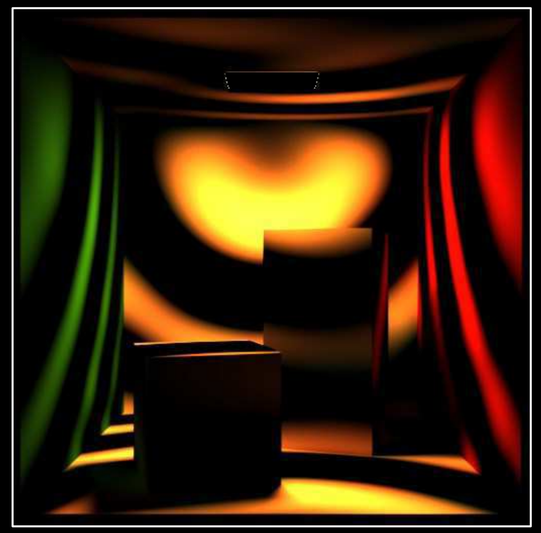
Path space integral for time-of-flight cameras

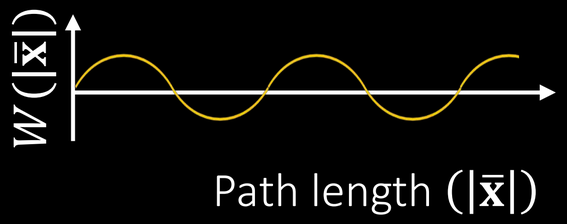
$$\text{image} = \int_{\text{light paths}} f(\text{path}) W(|\text{path}|)$$

intensity

BDPT, PT, PM, KDE, etc

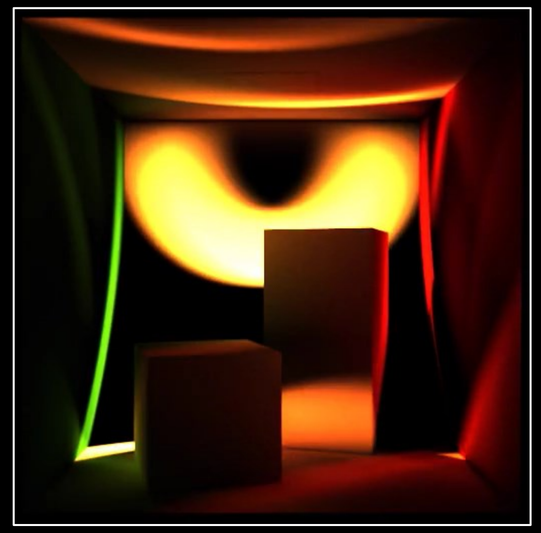
continuous-wave





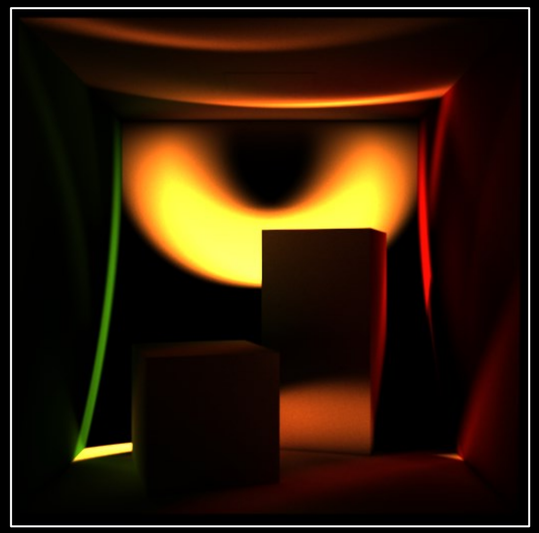
BDPT, PT, PM, KDE, etc

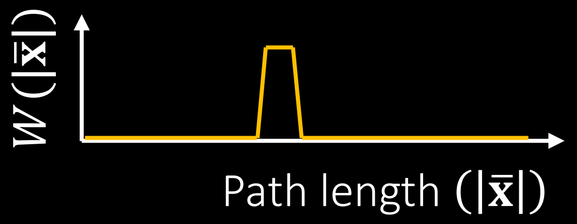
transient



BDPT, PT, PM, KDE, etc
[Jarabo et al., 2014, 2017]
[Marco et al. 2017, 2018]

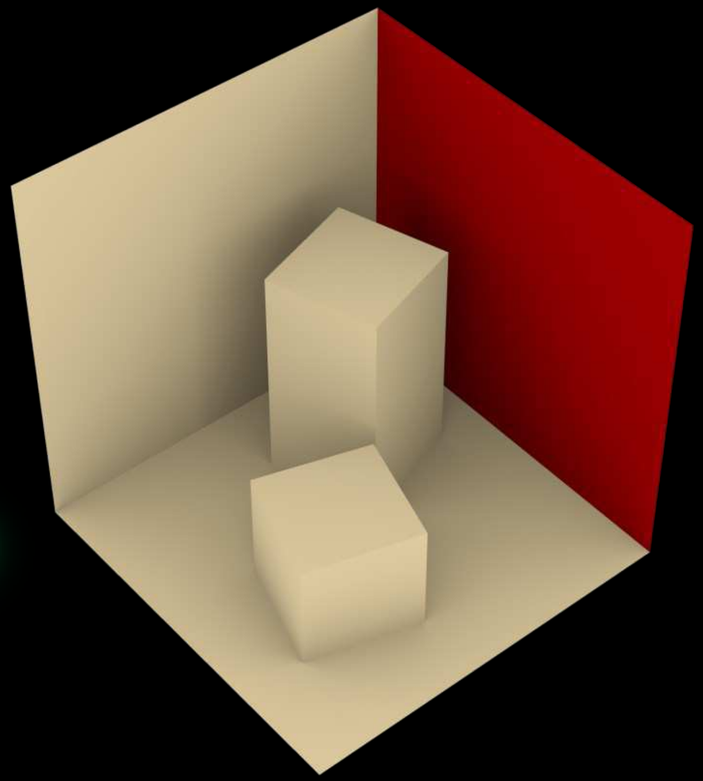
time-gated



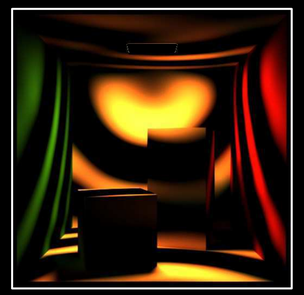


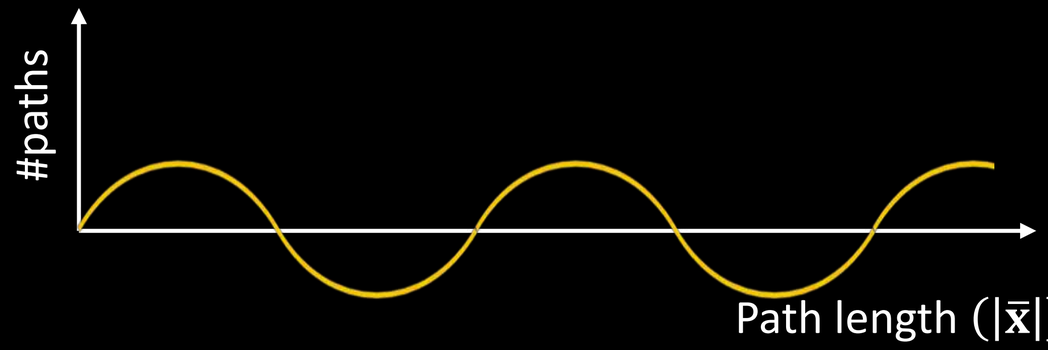
no efficient renderer

Path sampling for time-gated rendering is challenging



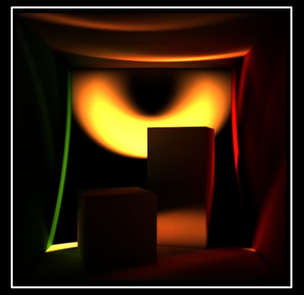
continuous-wave

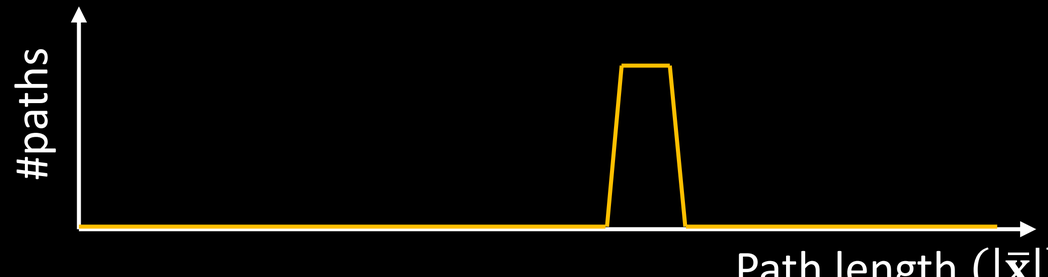




transient

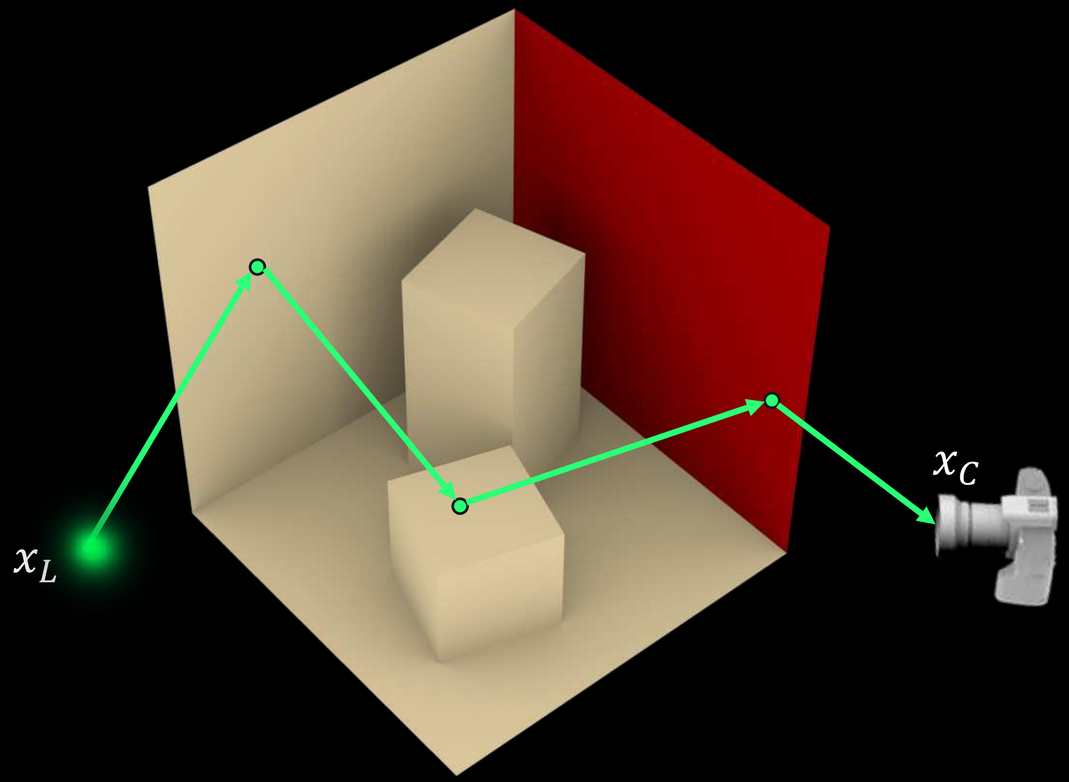
time-gated



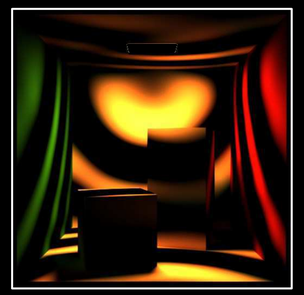


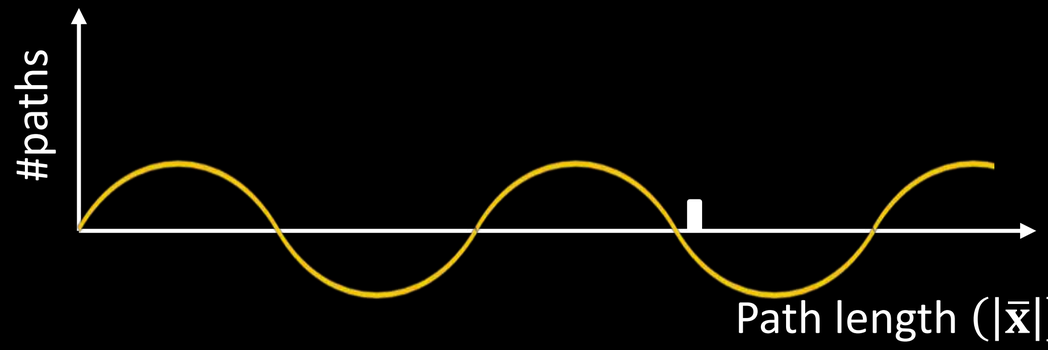
No control over path length

Path sampling for time-gated rendering is challenging



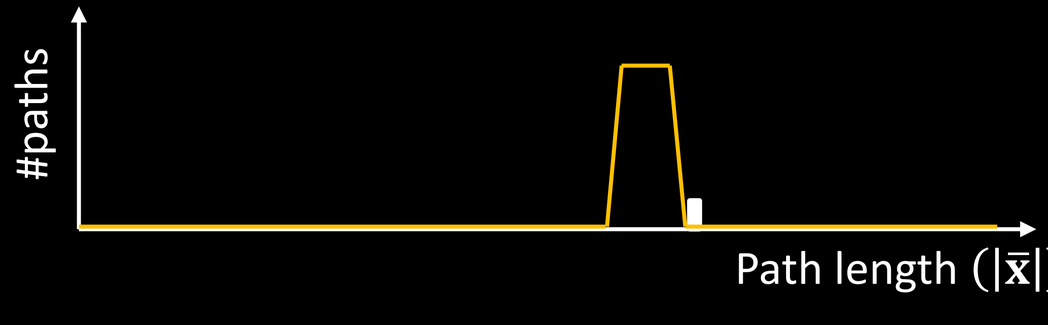
continuous-wave





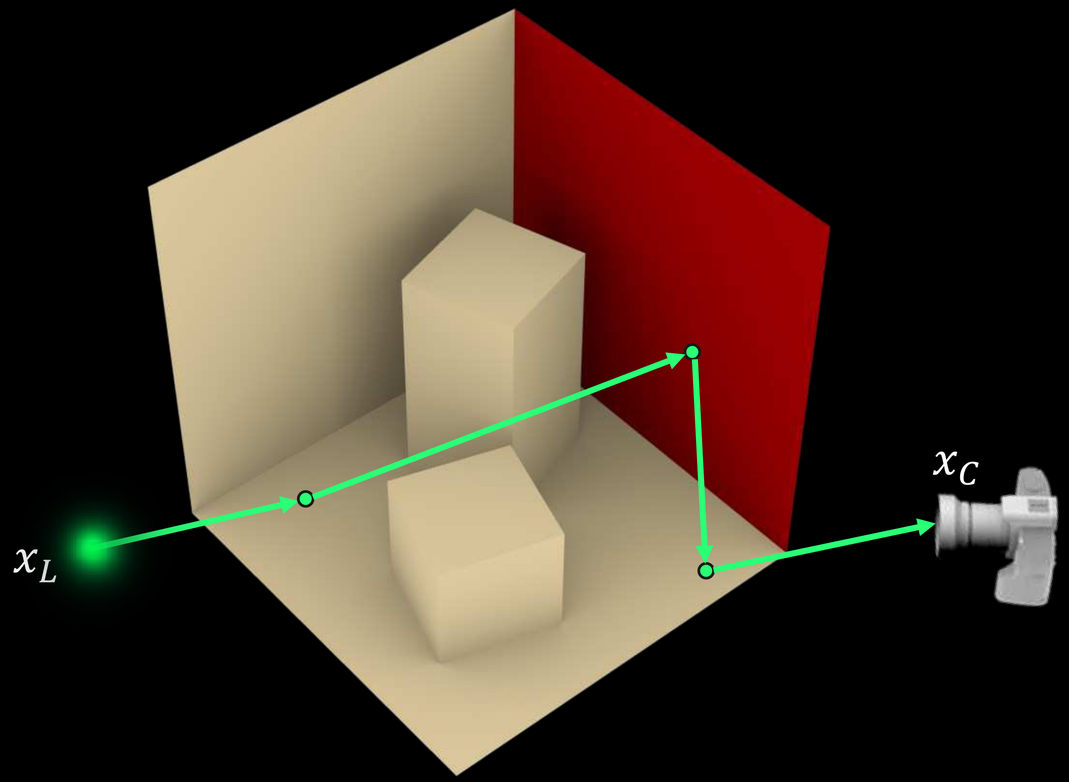
transient

time-gated

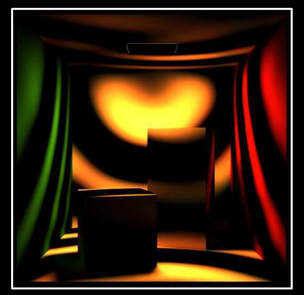


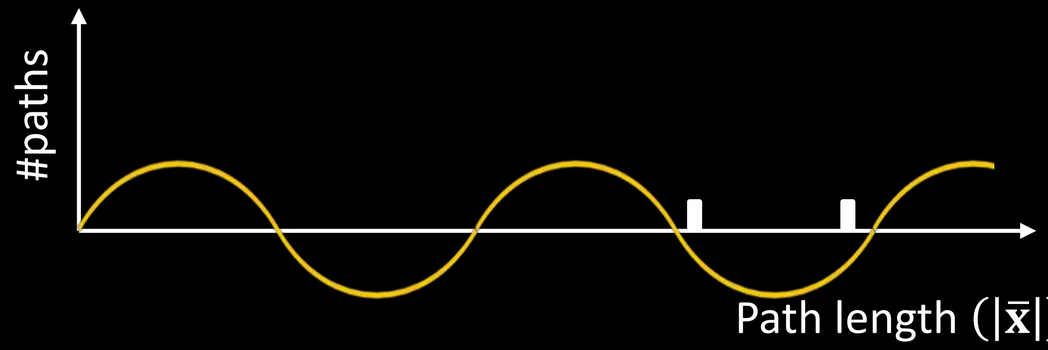
No control over path length

Path sampling for time-gated rendering is challenging



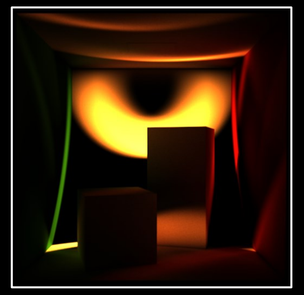
continuous-wave

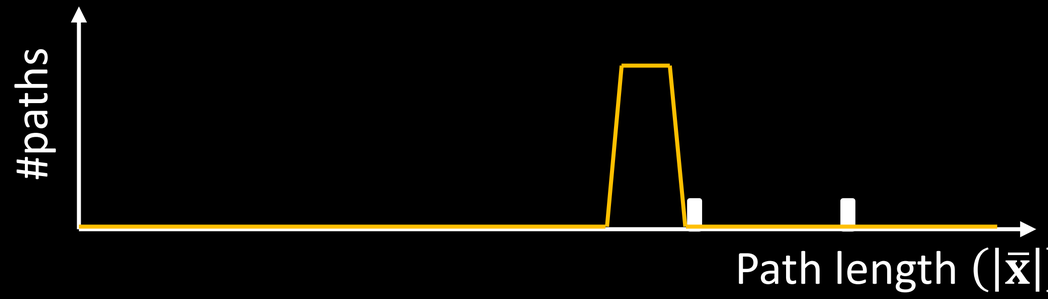




transient

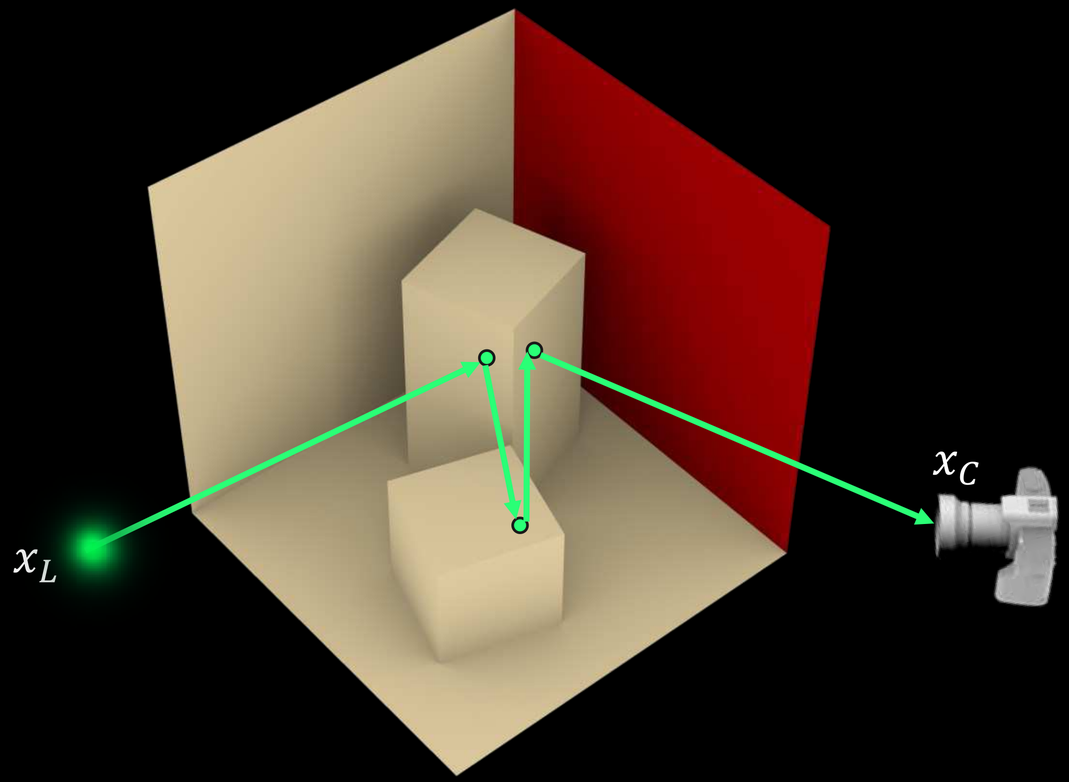
time-gated



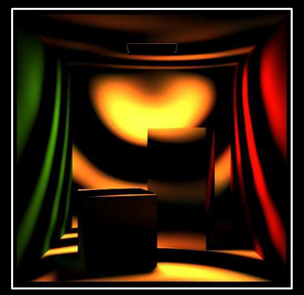


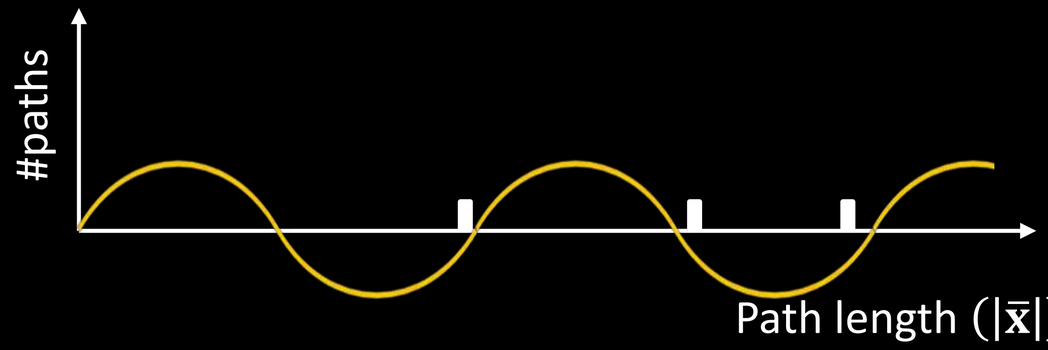
No control over path length

Path sampling for time-gated rendering is challenging

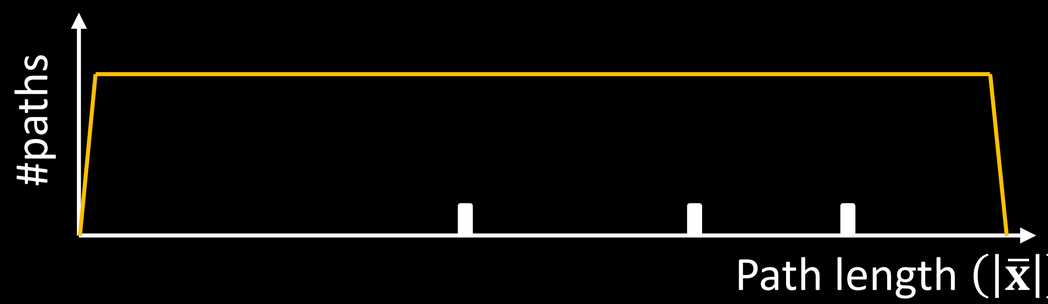


continuous-wave

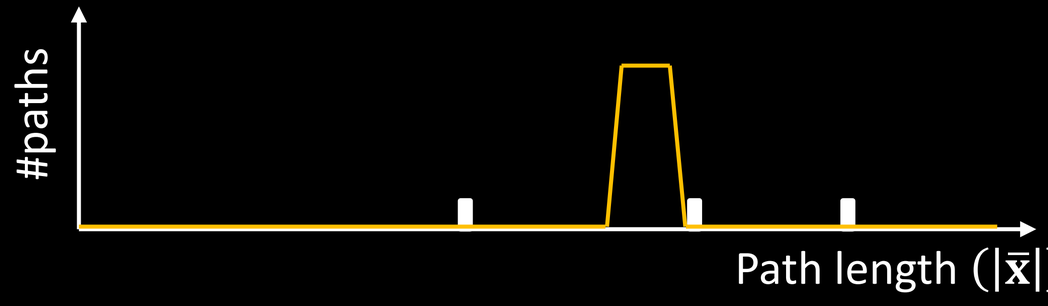




transient

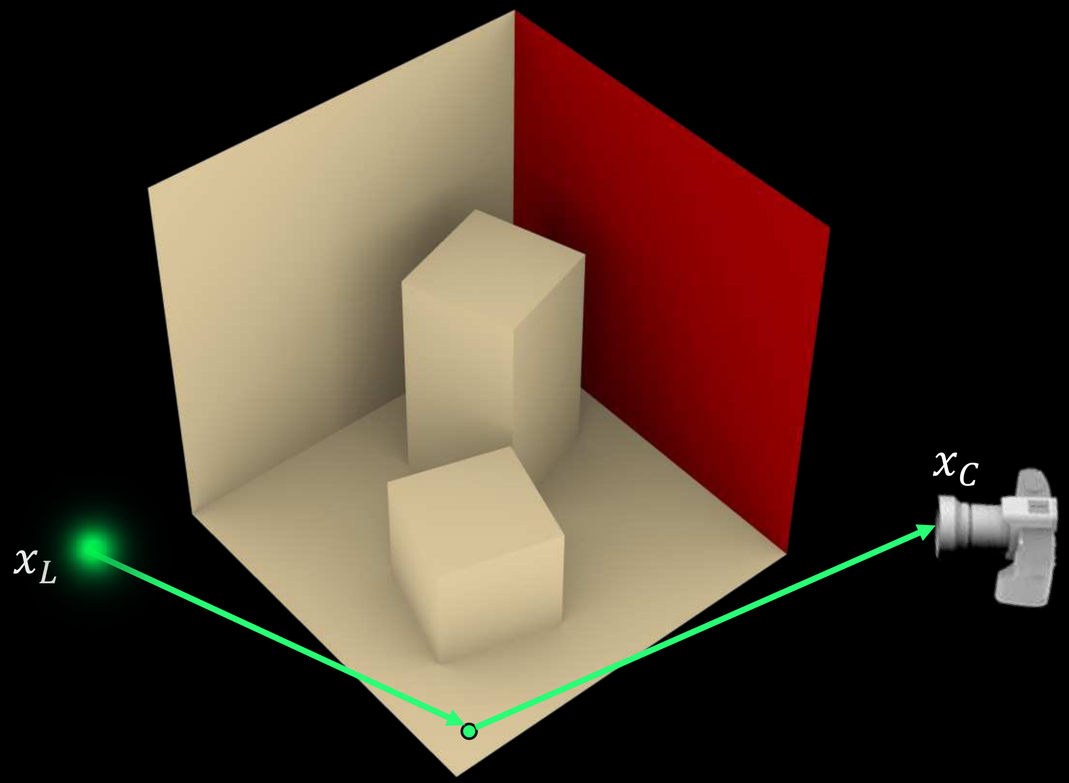


time-gated



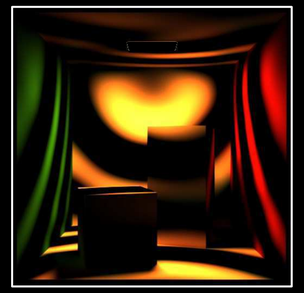
No control over path length

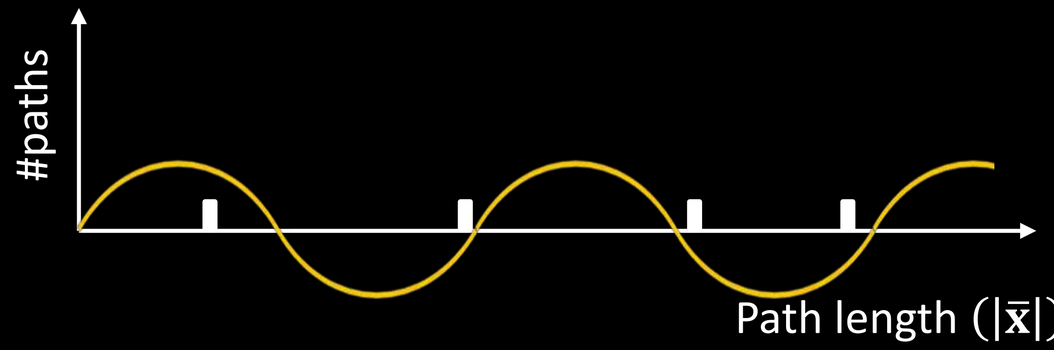
Path sampling for time-gated rendering is challenging



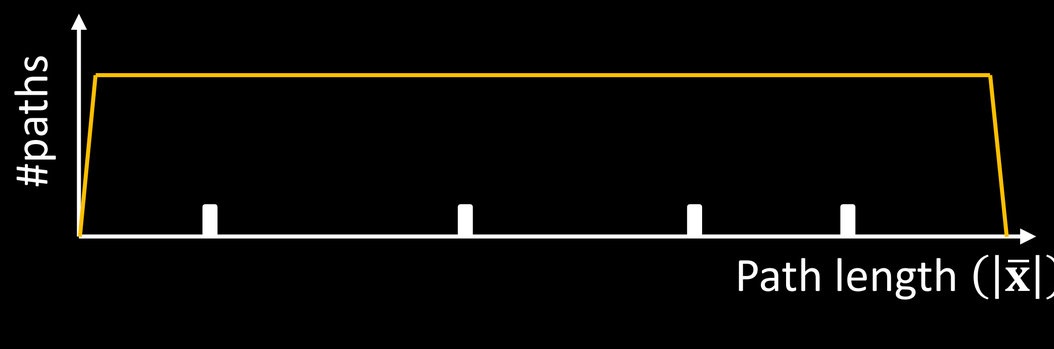
No control over path length

continuous-wave

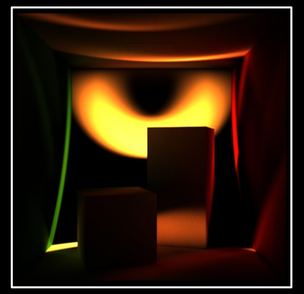


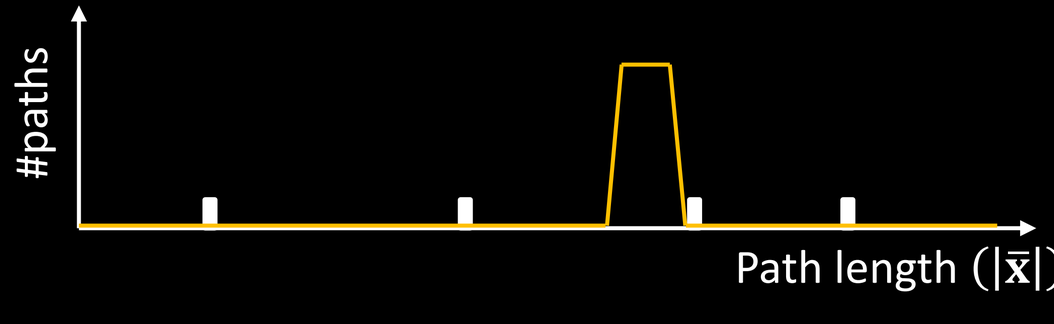


transient

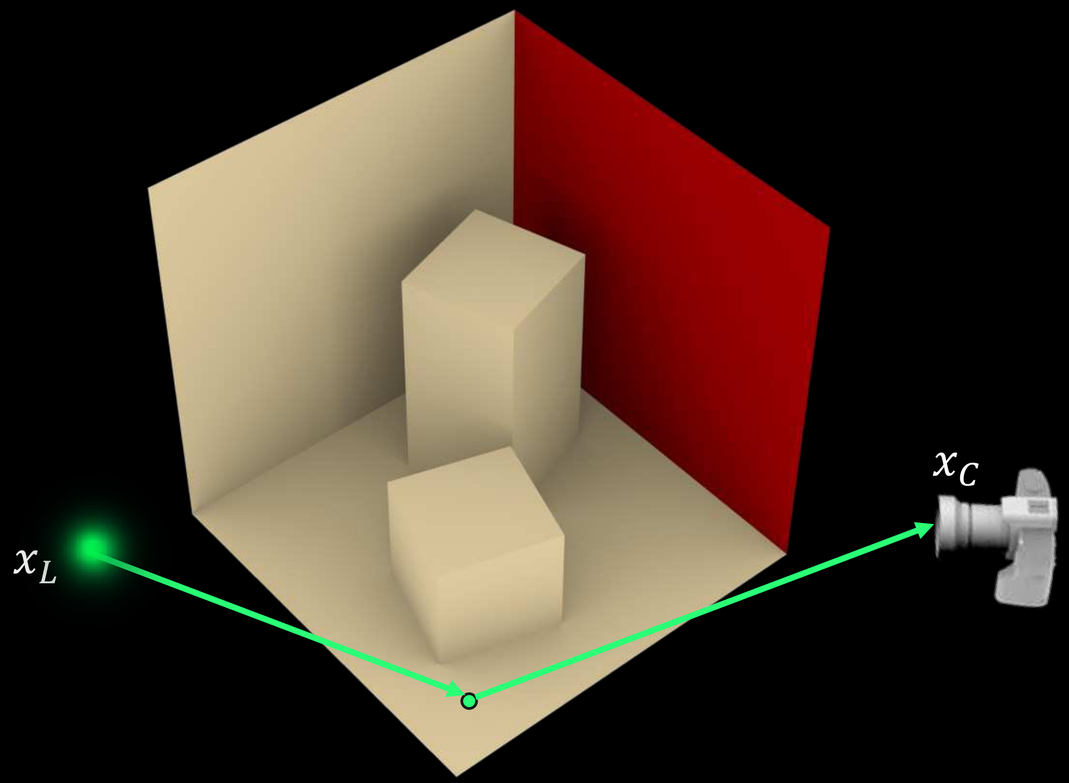


time-gated



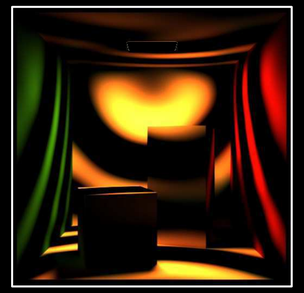


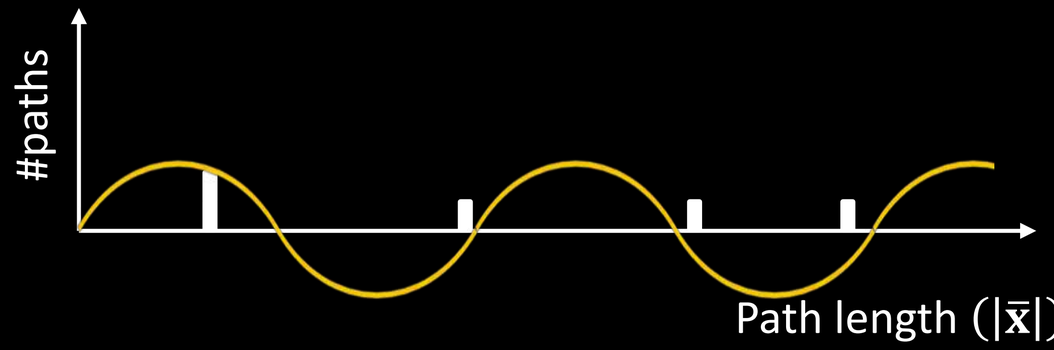
Path sampling for time-gated rendering is challenging



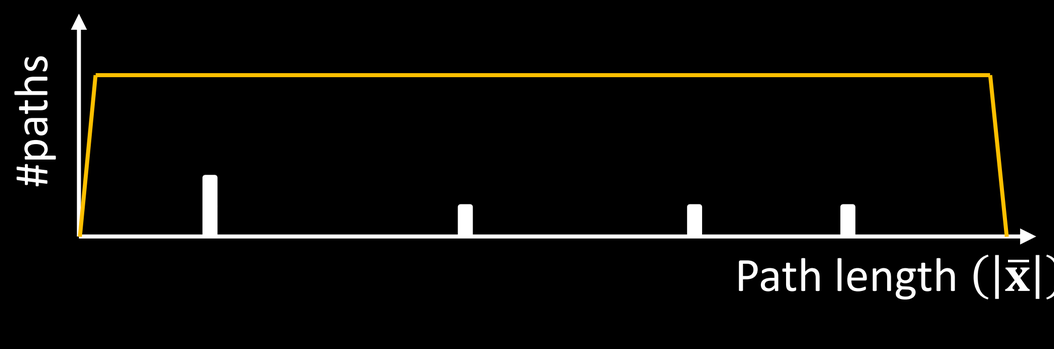
No control over path length

continuous-wave

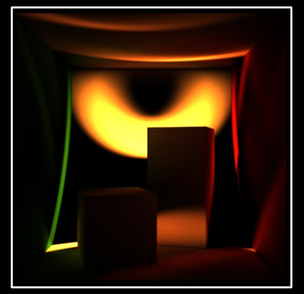


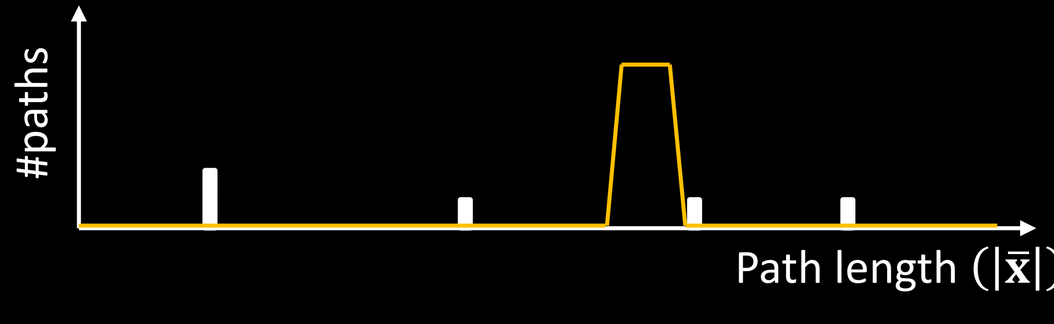


transient

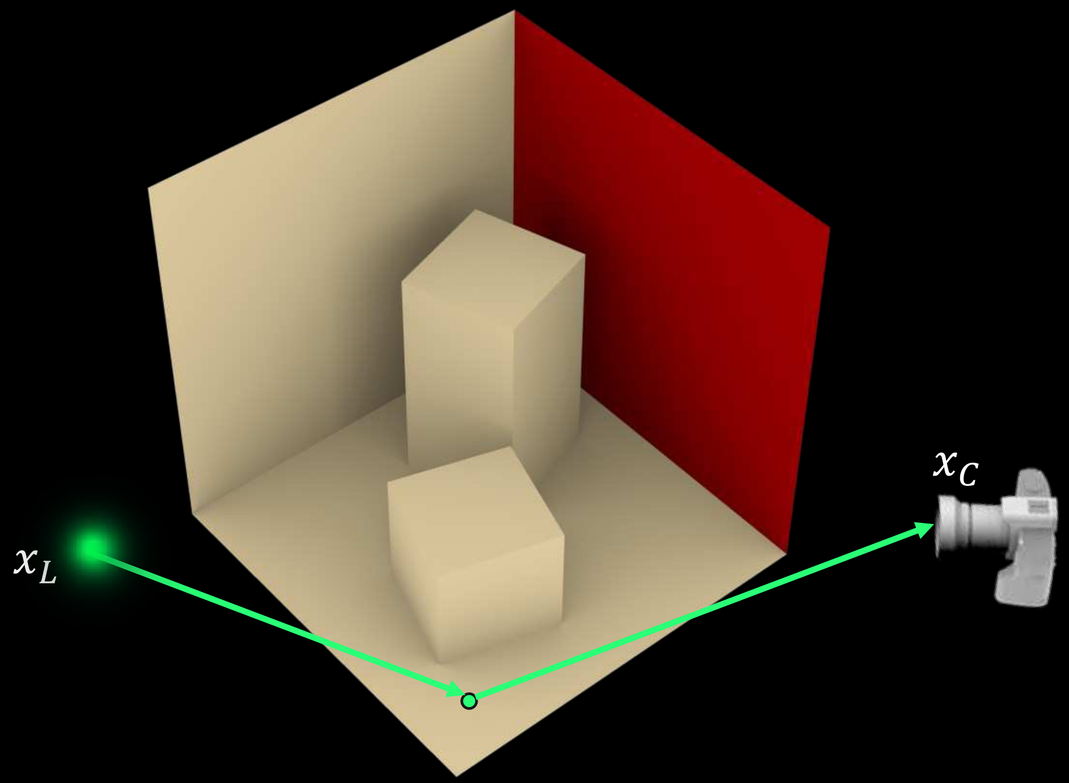


time-gated

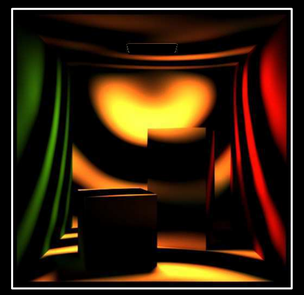


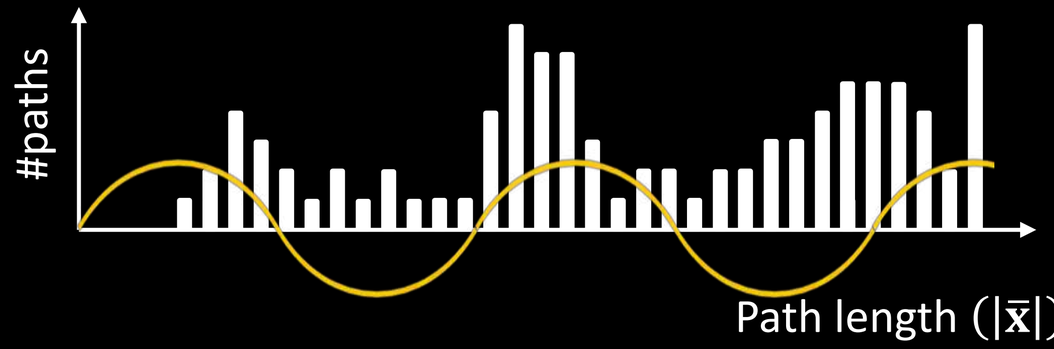


Path sampling for time-gated rendering is challenging

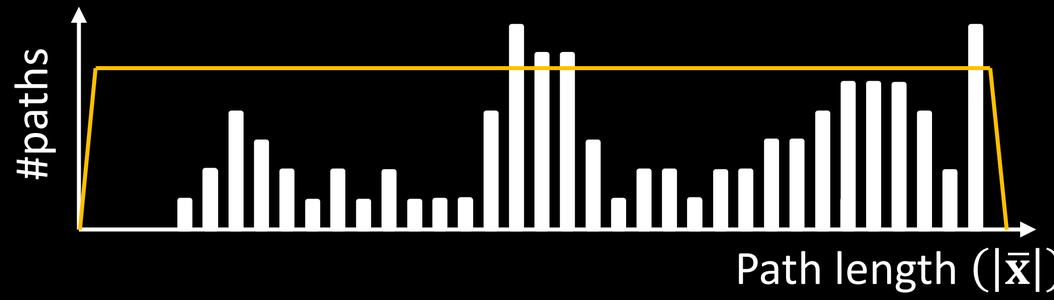


continuous-wave

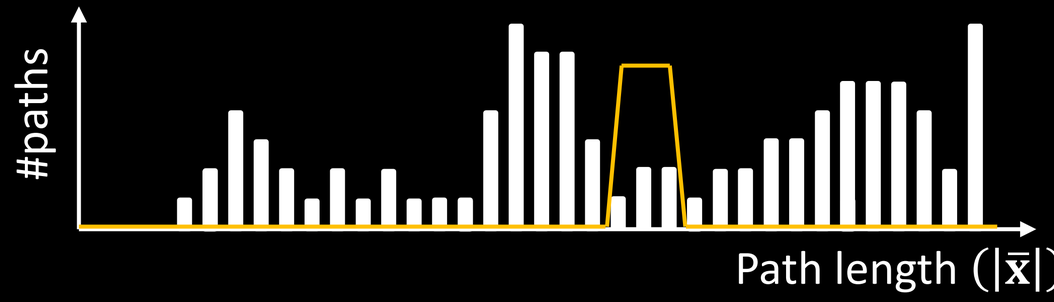




transient

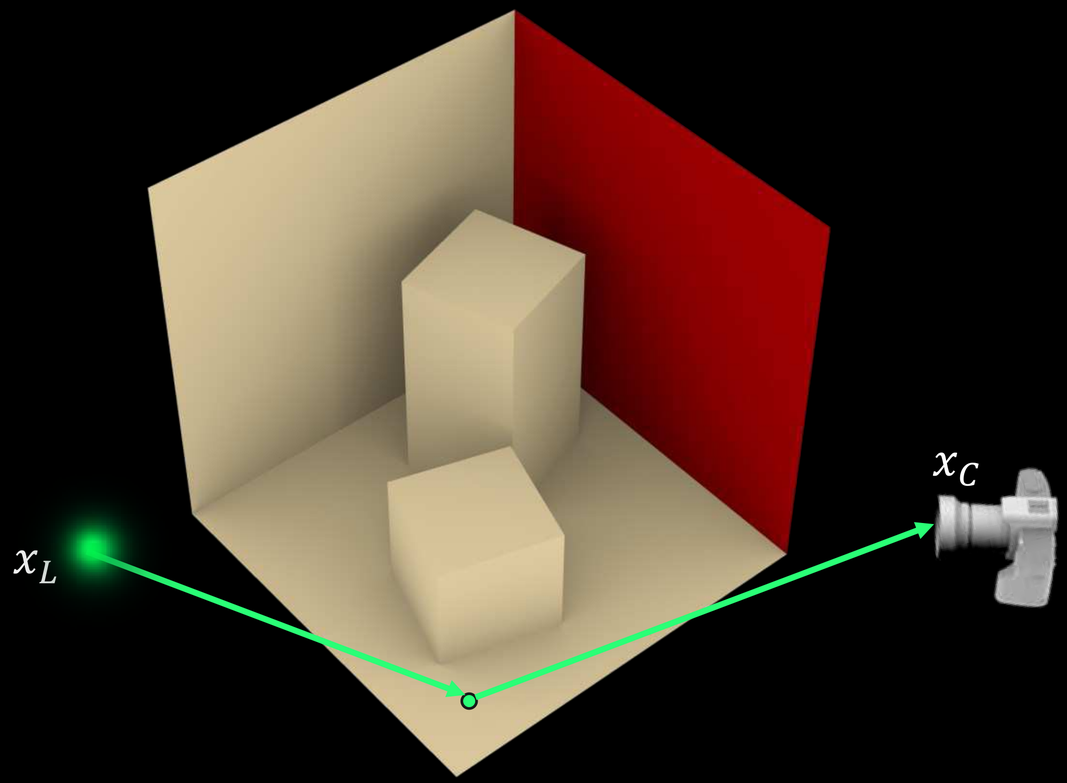


time-gated

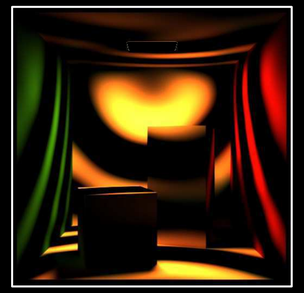


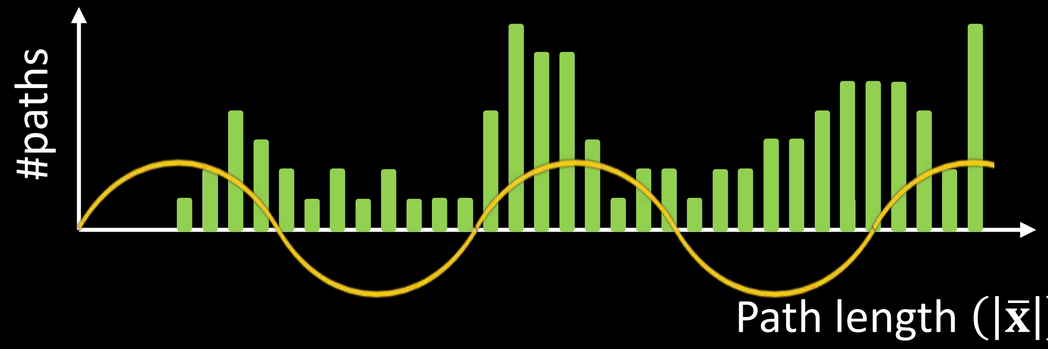
No control over path length

Path sampling for time-gated rendering is challenging

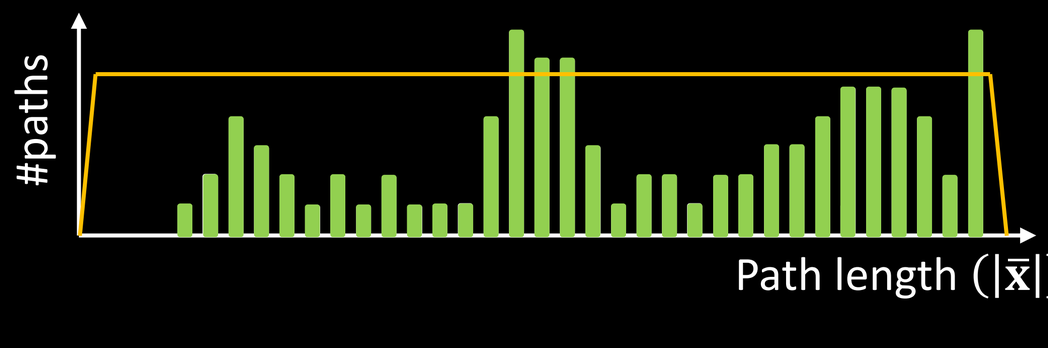


continuous-wave

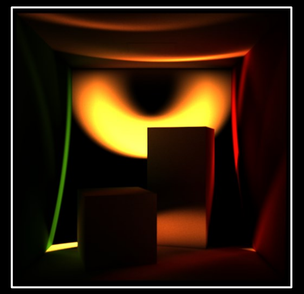


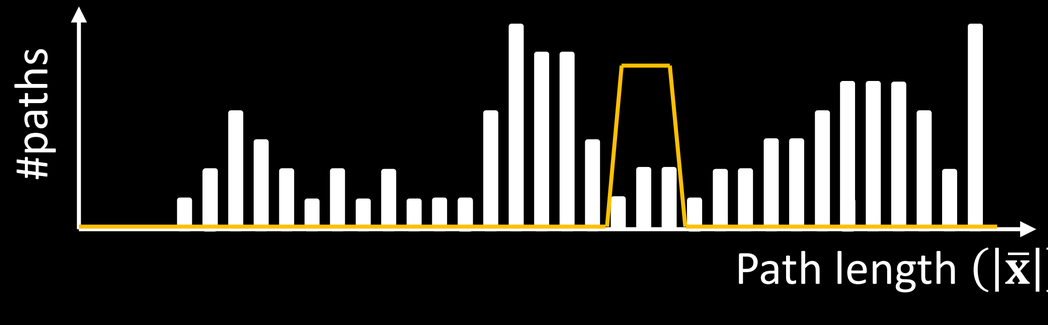


transient



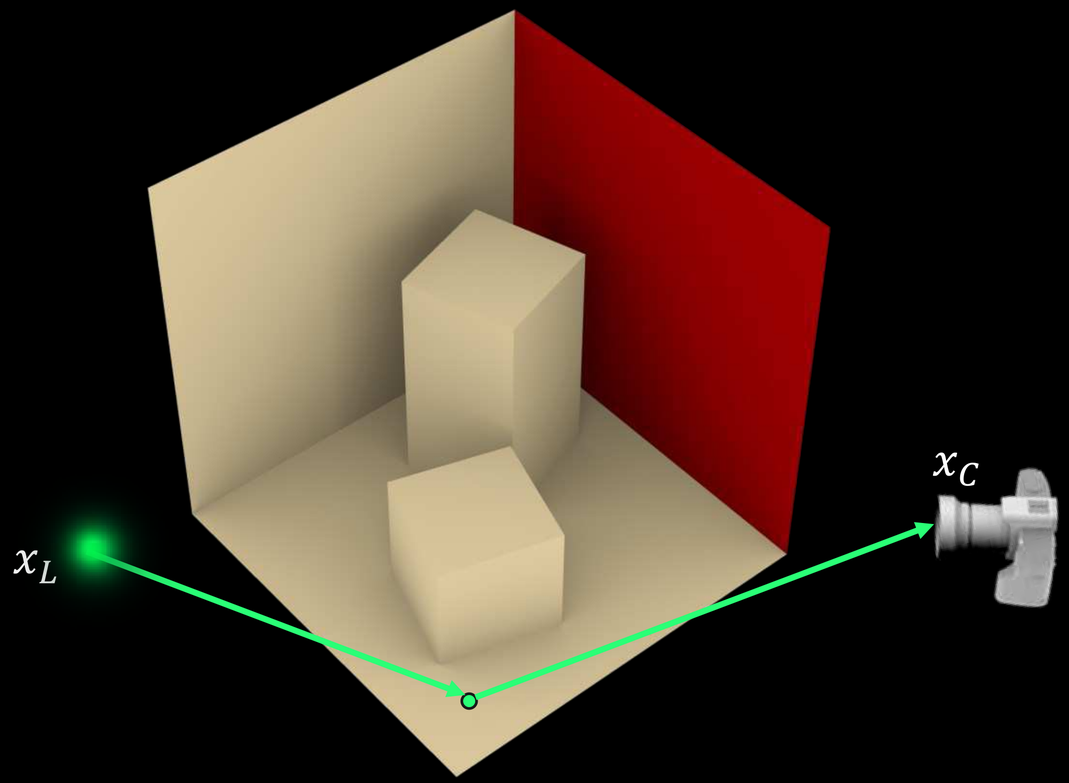
time-gated



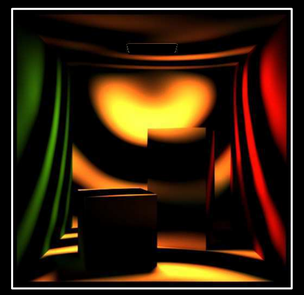


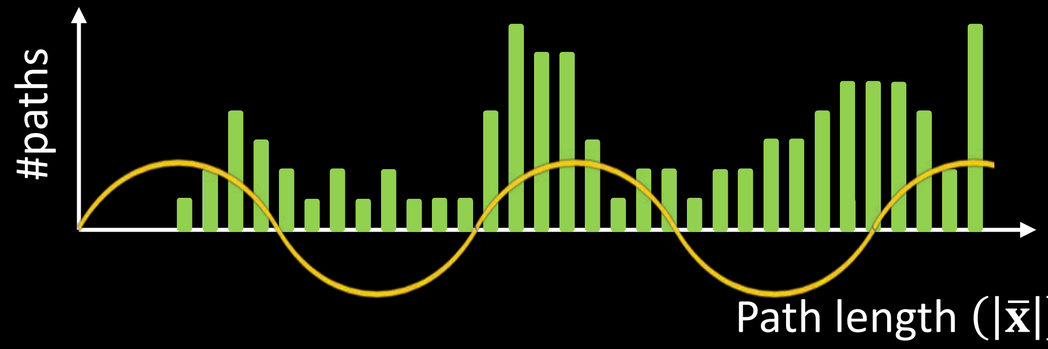
No control over path length

Path sampling for time-gated rendering is challenging

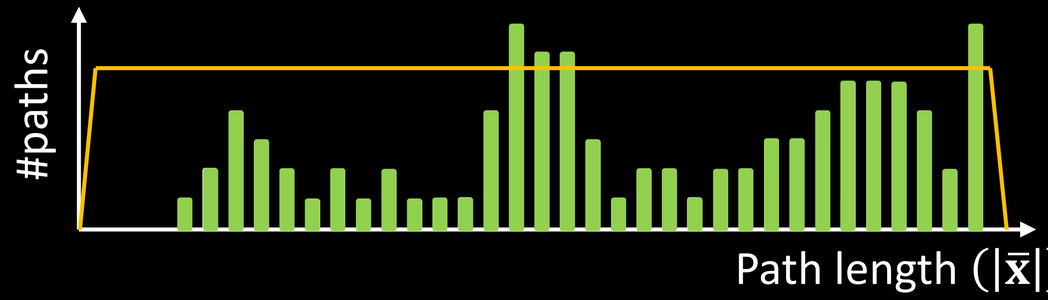


continuous-wave

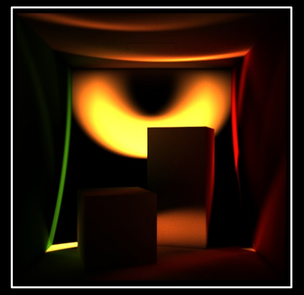


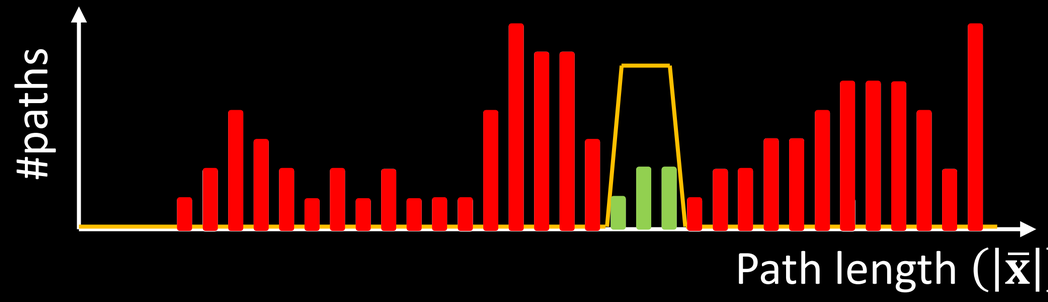


transient



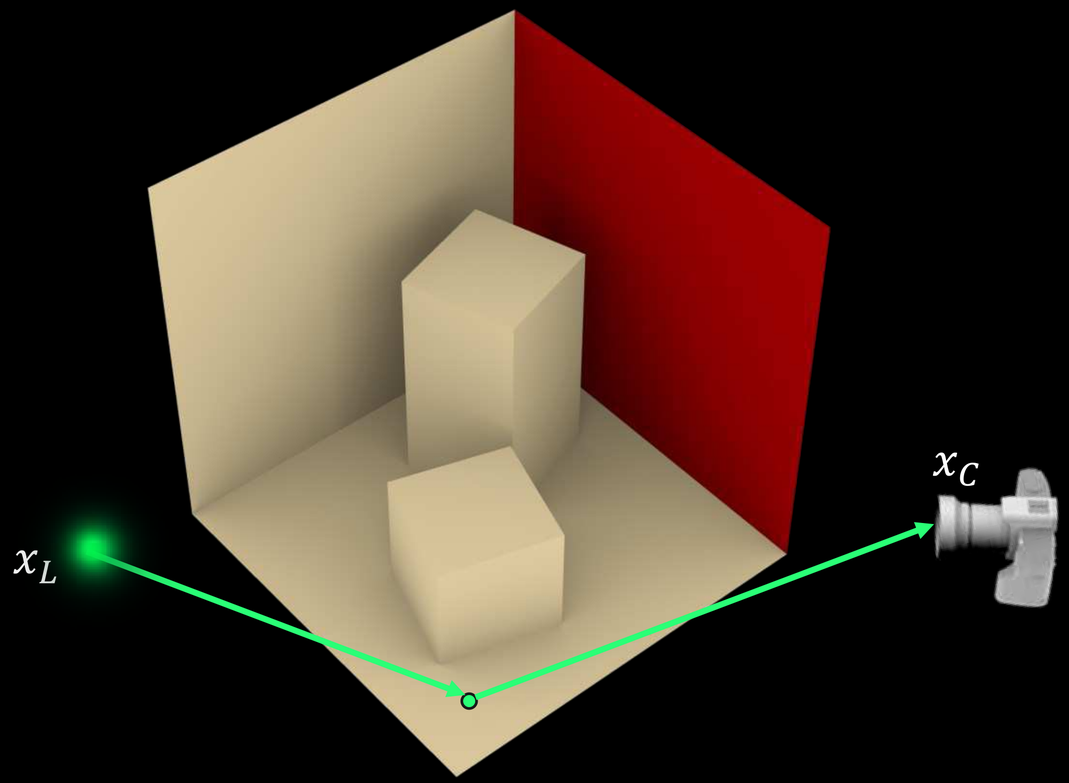
time-gated



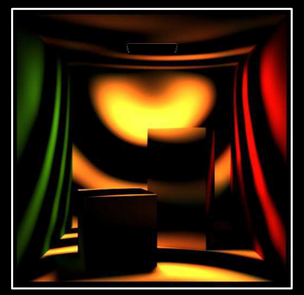


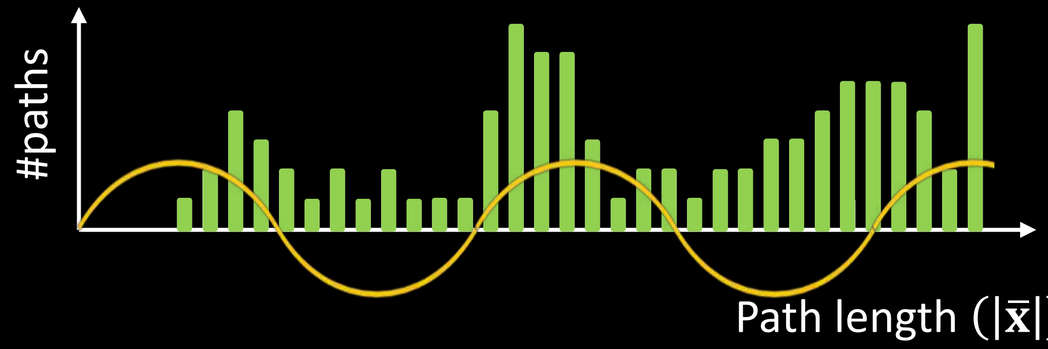
No control over path length

Path sampling for time-gated rendering is challenging

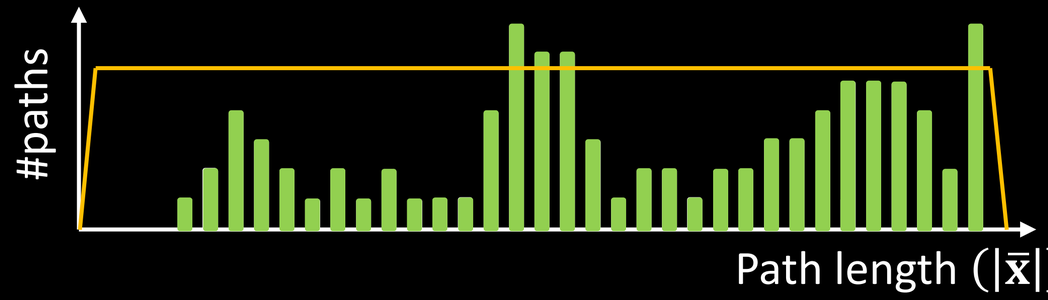


continuous-wave

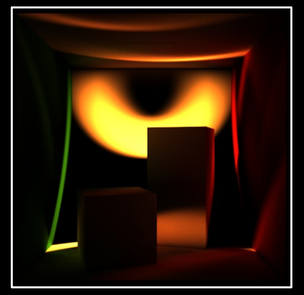


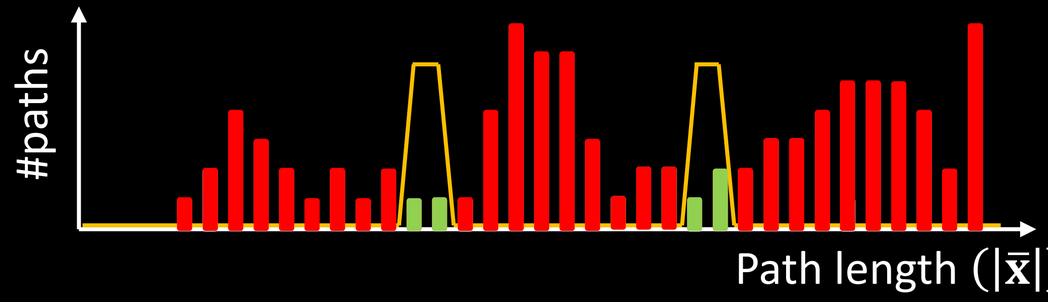


transient



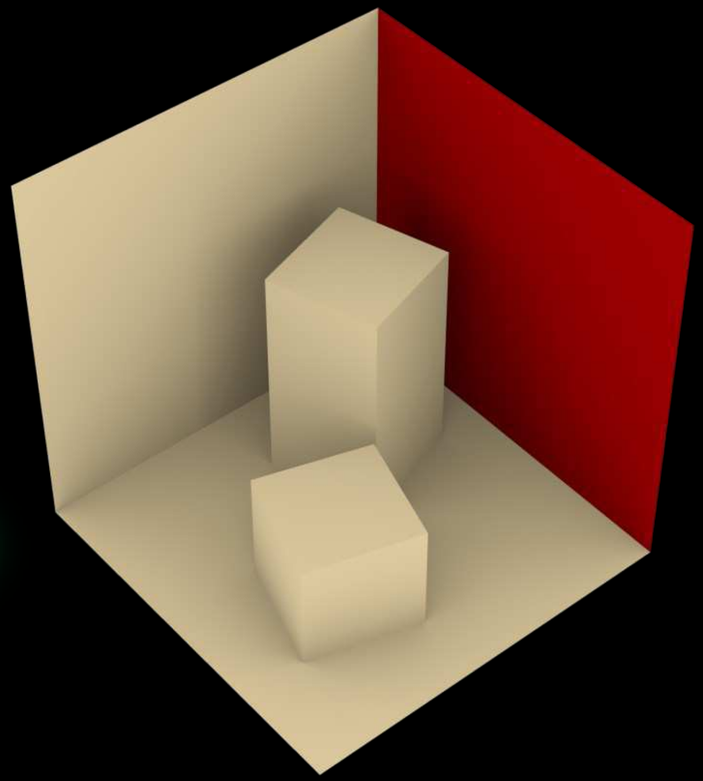
time-gated





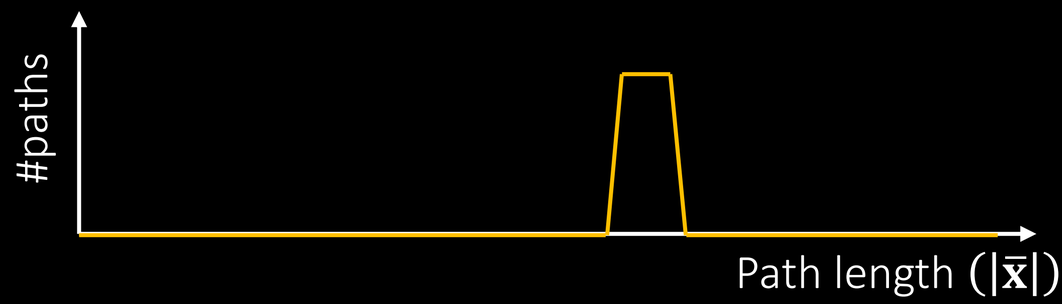
No control over path length

Path sampling for time-gated case

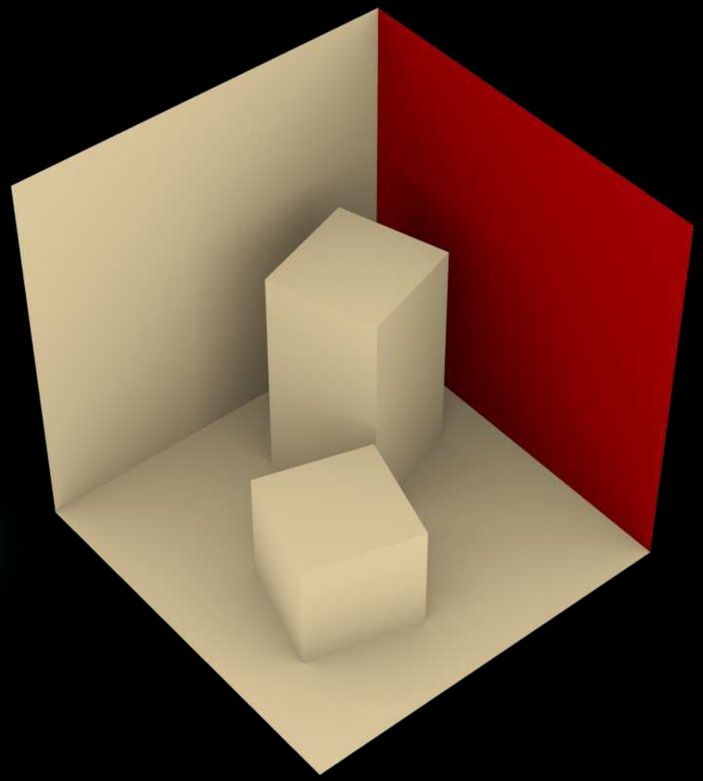


x_L

x_C

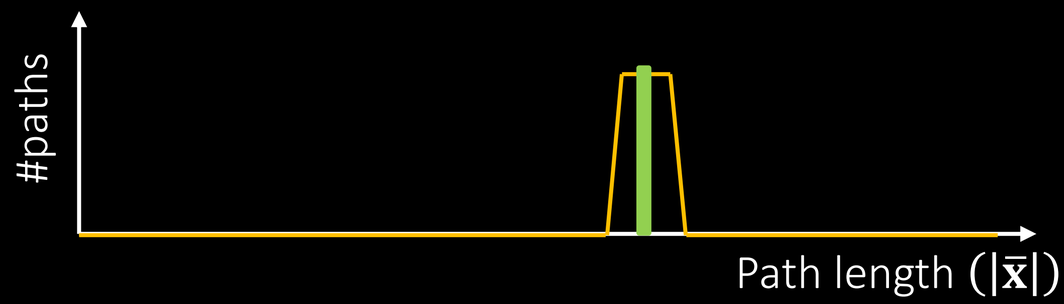


Path sampling for time-gated case



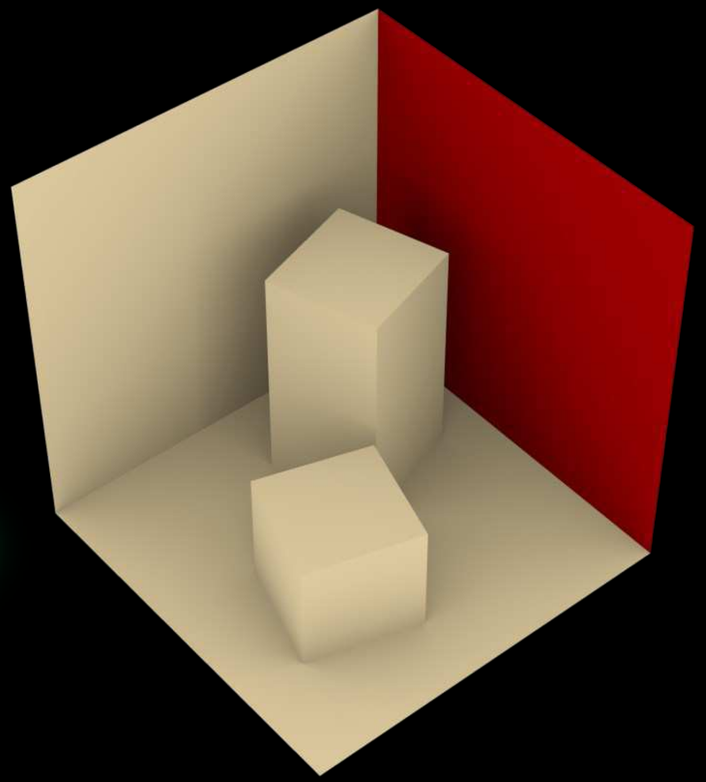
x_L

x_C



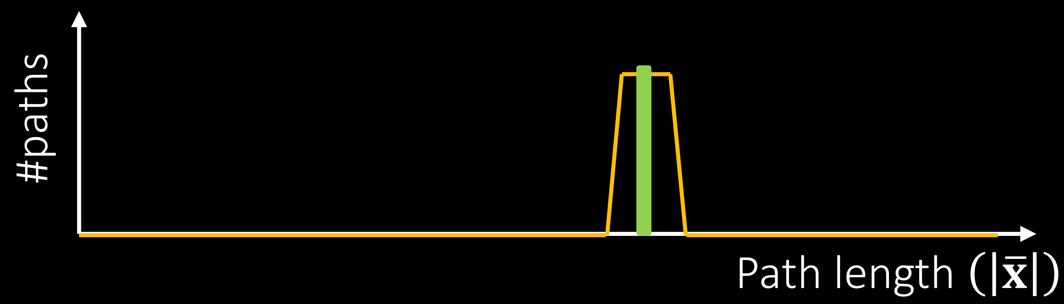
step 1: sample path length $|\bar{\mathbf{x}}|$

Path sampling for time-gated case



x_L

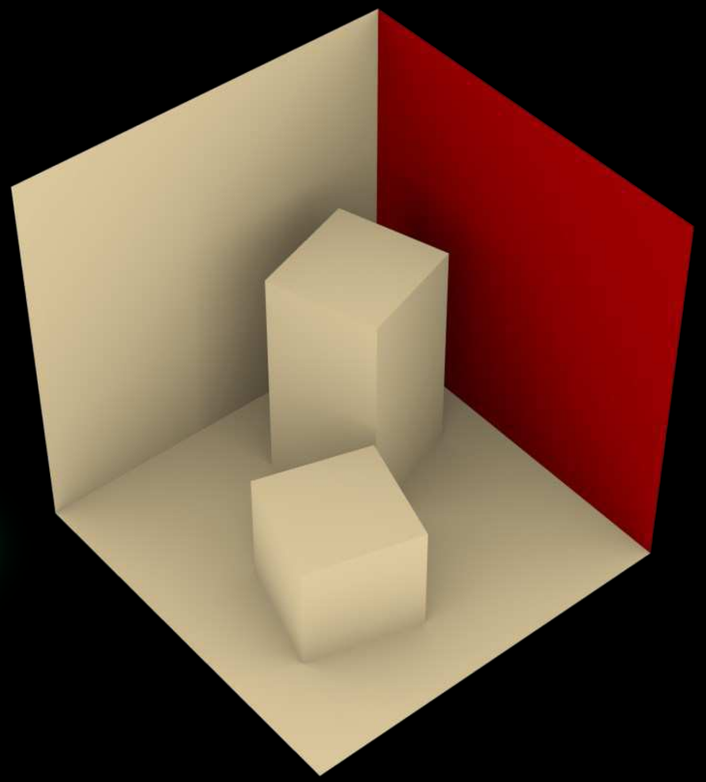
x_C



step 1: sample path length $|\bar{\mathbf{x}}|$

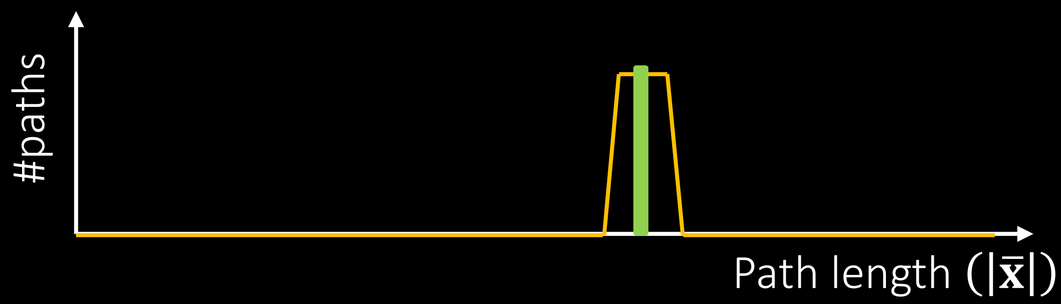
step 2: generate path with target length $|\bar{\mathbf{x}}|$

Path sampling for time-gated case



x_L

x_C

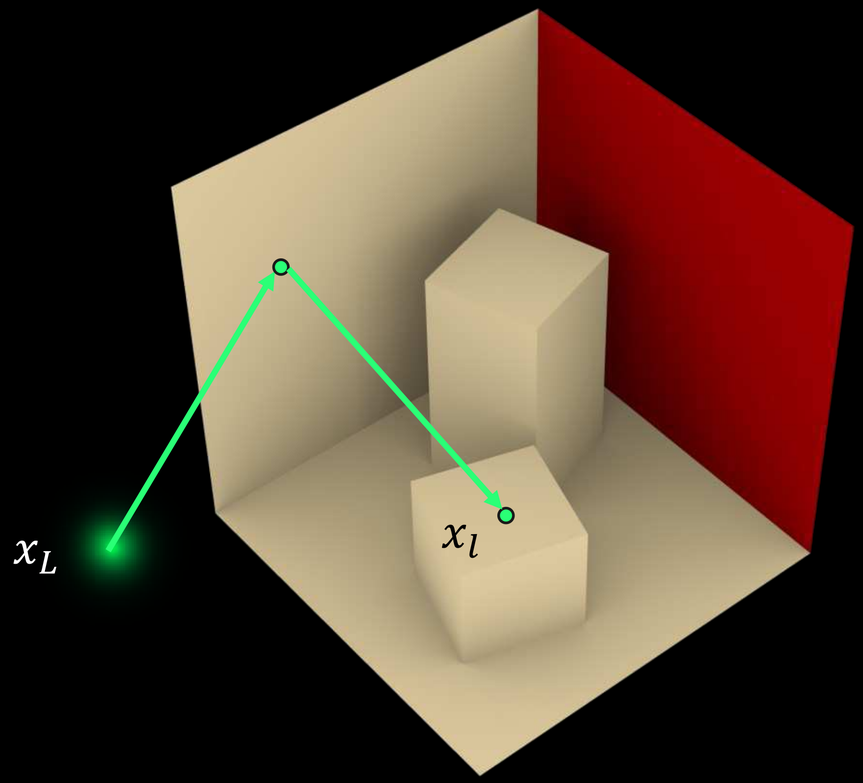


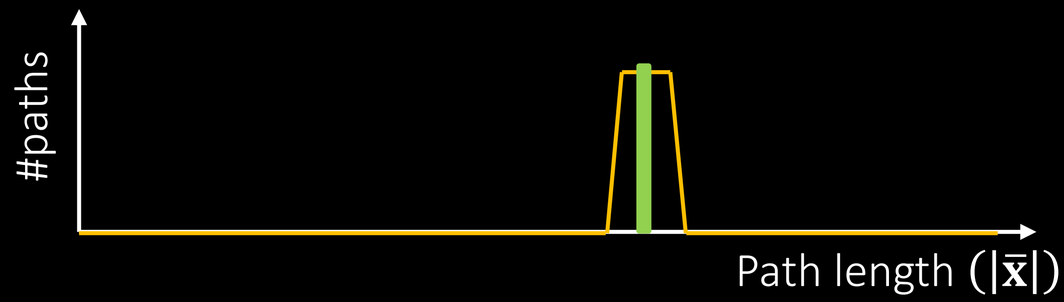
step 1: sample path length $|\bar{\mathbf{x}}|$

step 2: generate path with target length $|\bar{\mathbf{x}}|$

bidirectional path tracing (BDPT)

Path sampling for time-gated case





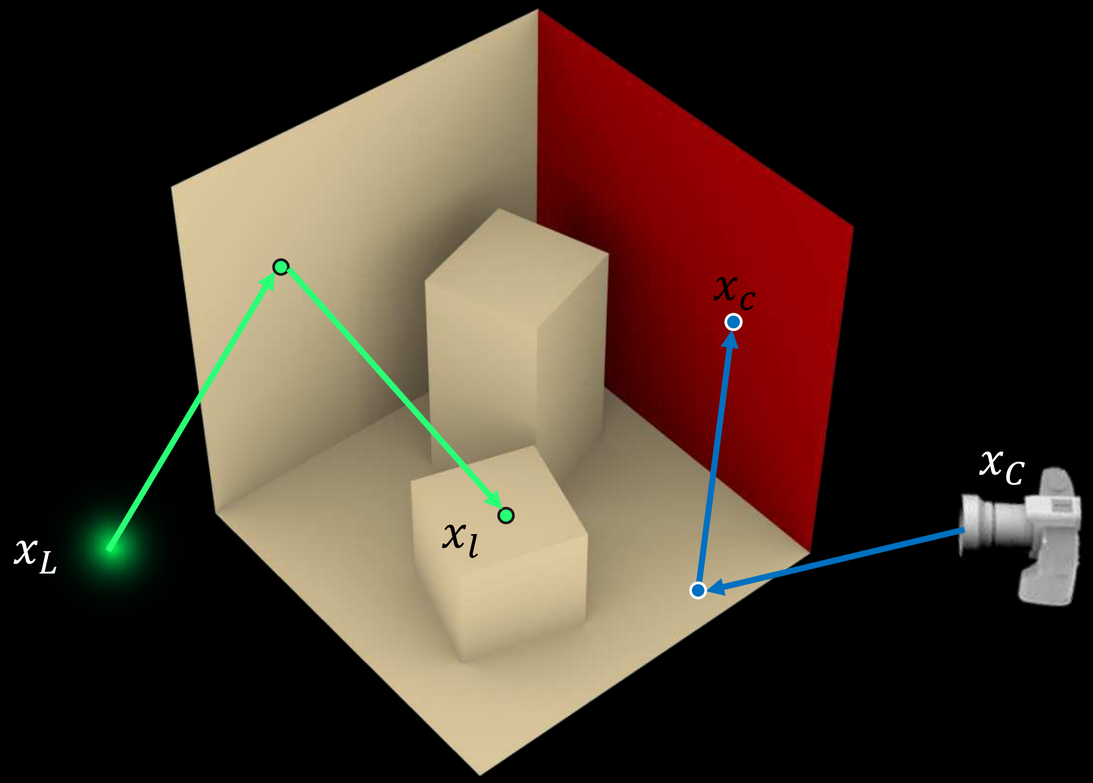
step 1: sample path length $|\bar{\mathbf{x}}|$

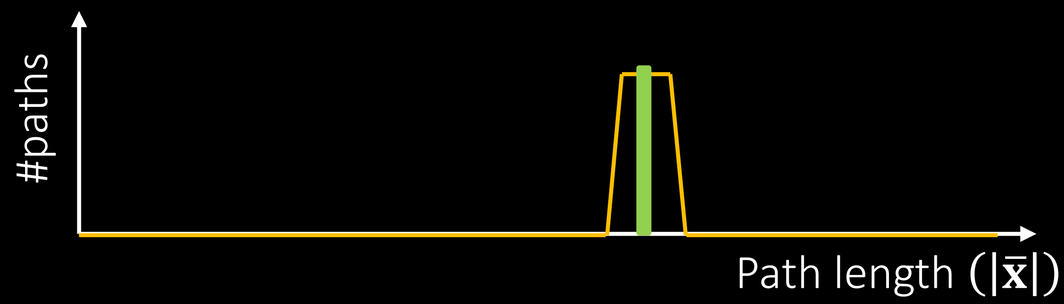
step 2: generate path with target length $|\bar{\mathbf{x}}|$

bidirectional path tracing (BDPT)

generate light sub-path, camera sub-path

Path sampling for time-gated case





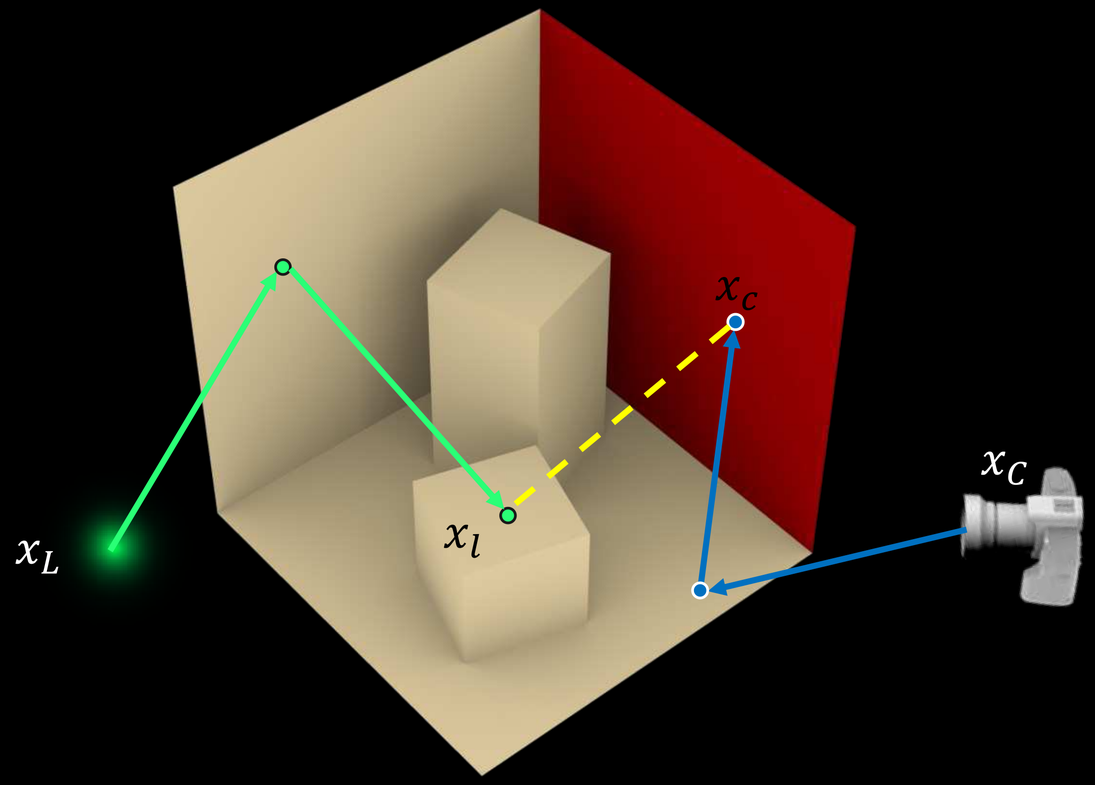
step 1: sample path length $|\bar{\mathbf{x}}|$

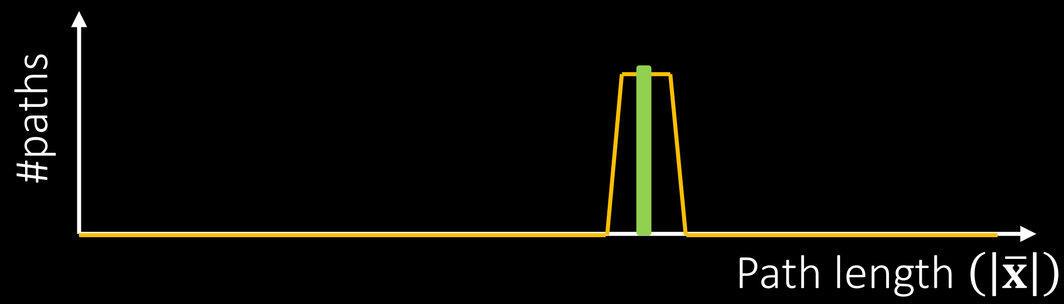
step 2: generate path with target length $|\bar{\mathbf{x}}|$

bidirectional path tracing (BDPT)

generate light sub-path, camera sub-path

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

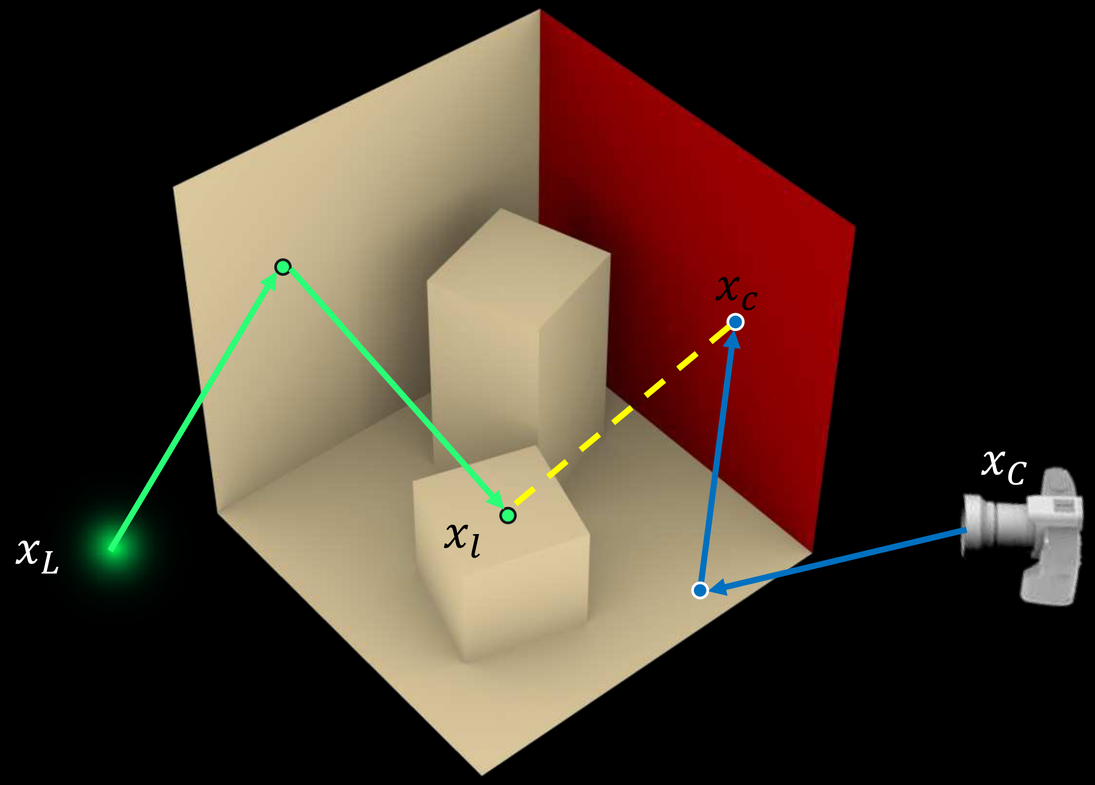
step 2: generate path with target length $|\bar{\mathbf{x}}|$

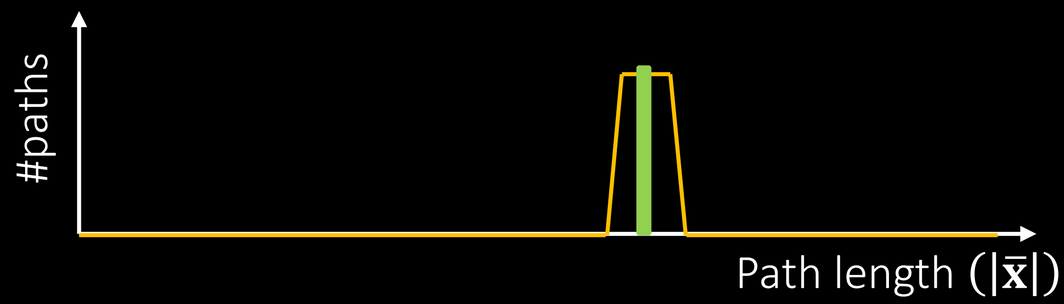
bidirectional path tracing (BDPT)

generate light sub-path, camera sub-path

join source sub-path end x_s and camera sub-path end x_c

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

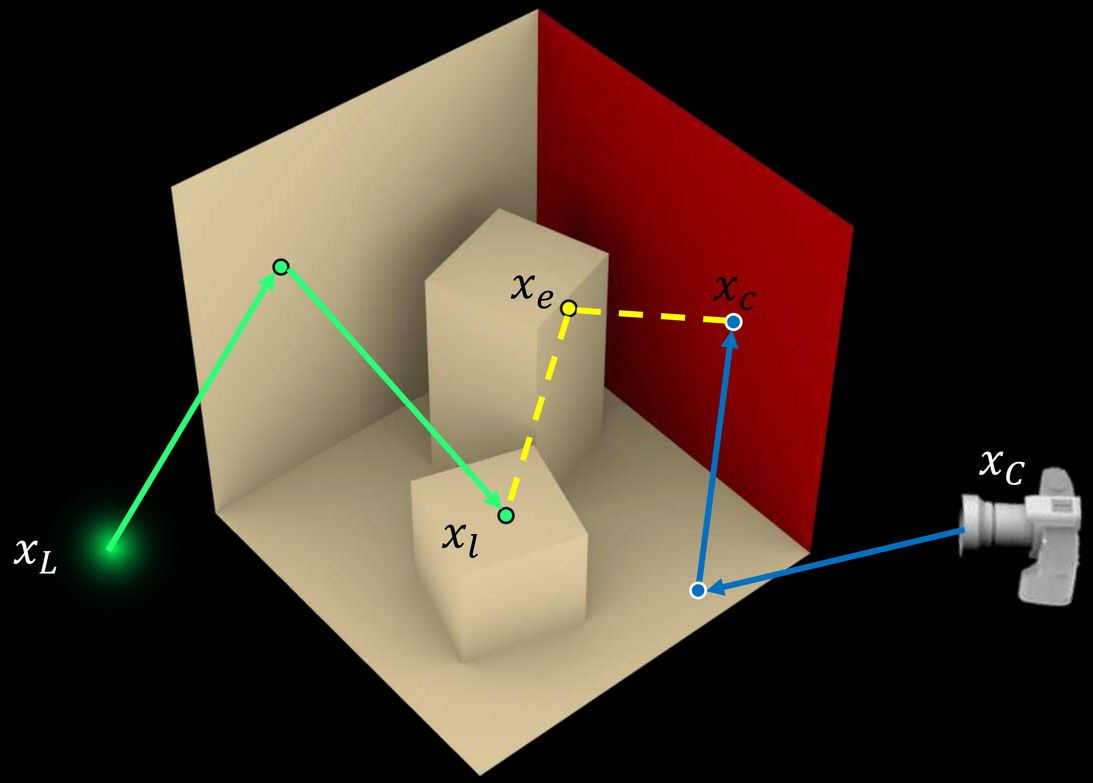
step 2: generate path with target length $|\bar{\mathbf{x}}|$

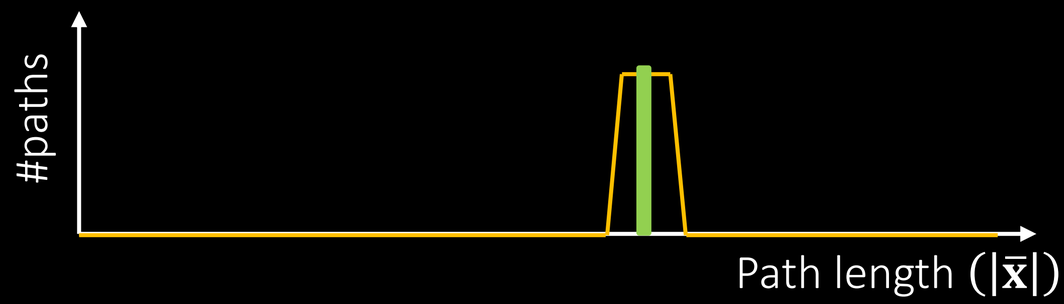
bidirectional path tracing (BDPT)

generate light sub-path, camera sub-path

~~join source sub-path end x_s and camera sub-path end x_c~~

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

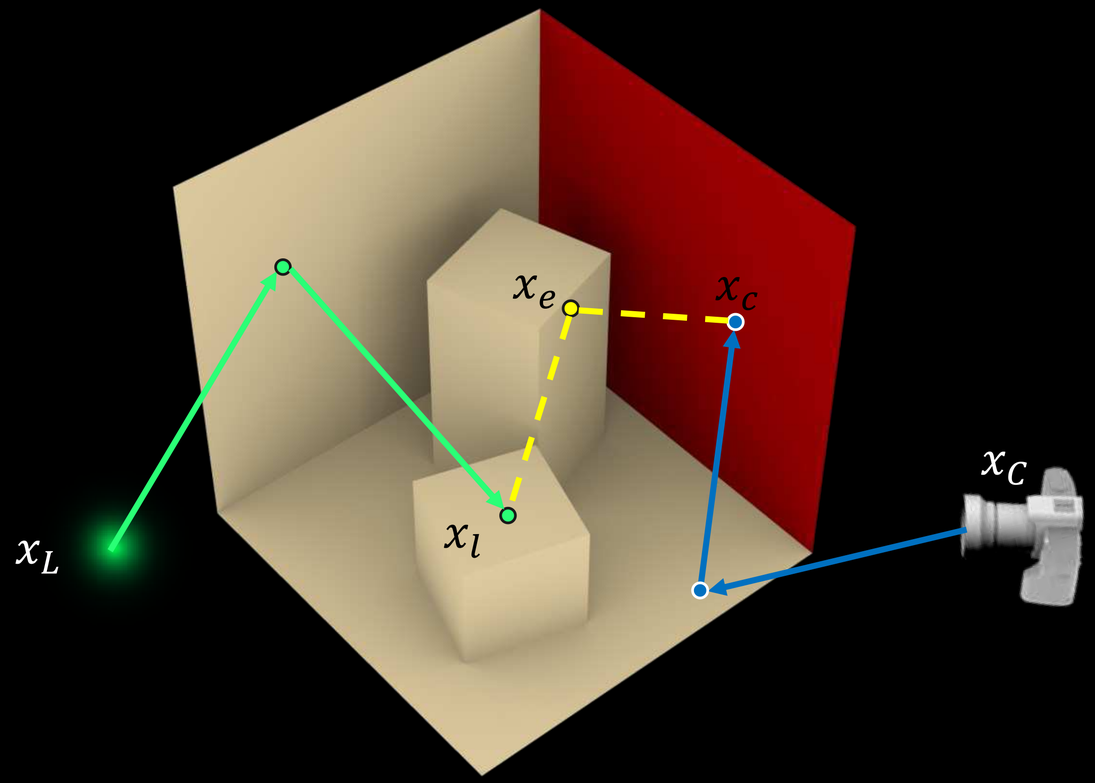
step 2: generate path with target length $|\bar{\mathbf{x}}|$

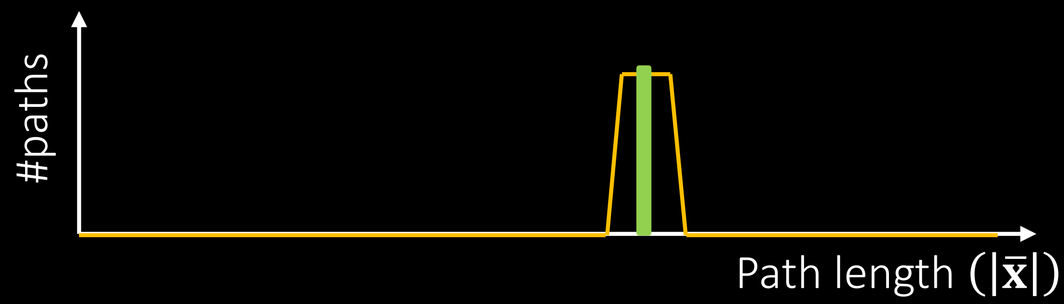
bidirectional path tracing (BDPT)

generate light sub-path, camera sub-path

join x_l and x_c via connecting vertex (x_e)

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

step 2: generate path with target length $|\bar{\mathbf{x}}|$

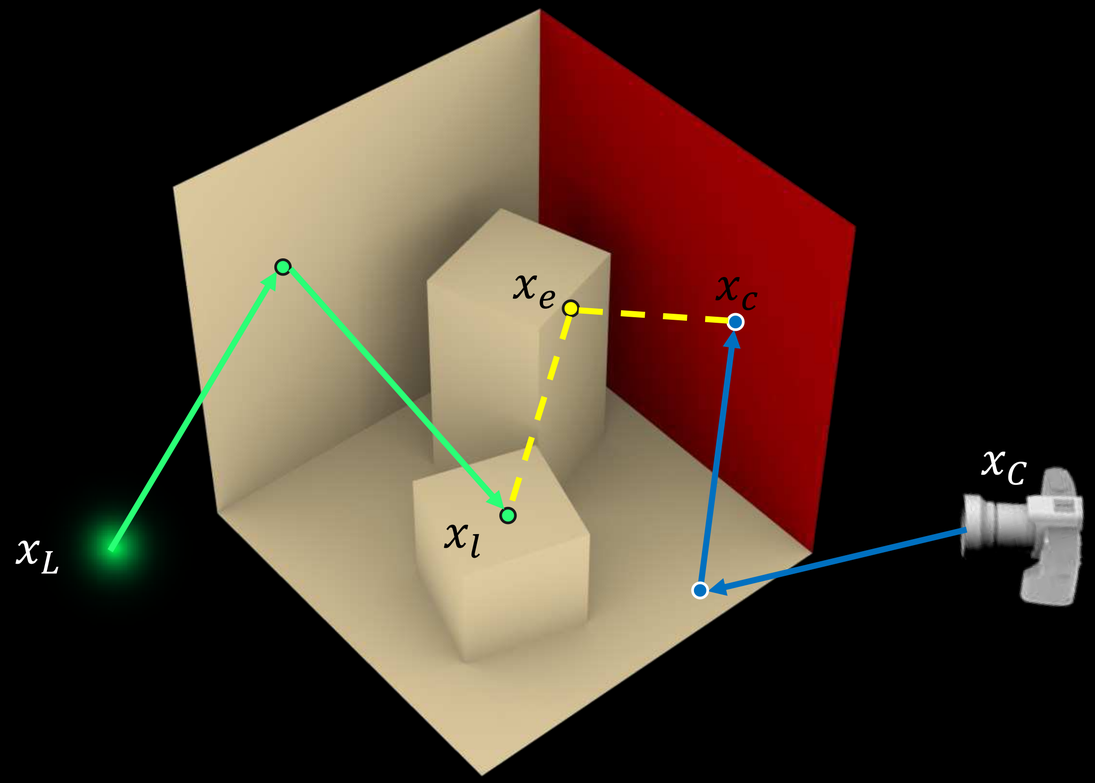
bidirectional path tracing (BDPT)

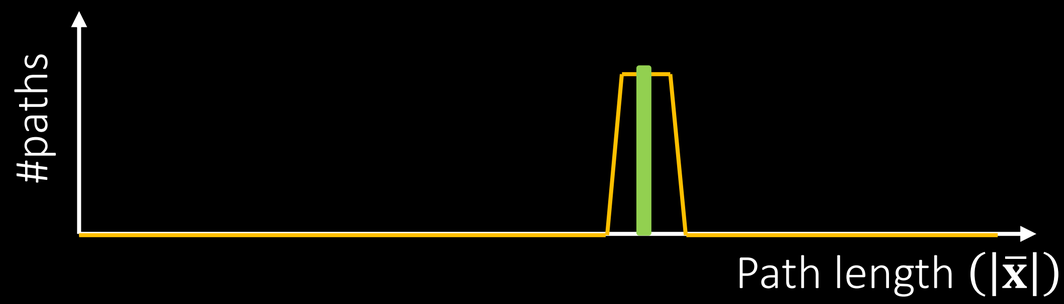
generate light sub-path, camera sub-path

join x_l and x_c via connecting vertex (x_e)

source sub–path length +
sensor sub–path length +
$|x_e \rightarrow x_l| + |x_e \rightarrow x_c|$
$= |\bar{\mathbf{x}}|$

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

step 2: generate path with target length $|\bar{\mathbf{x}}|$

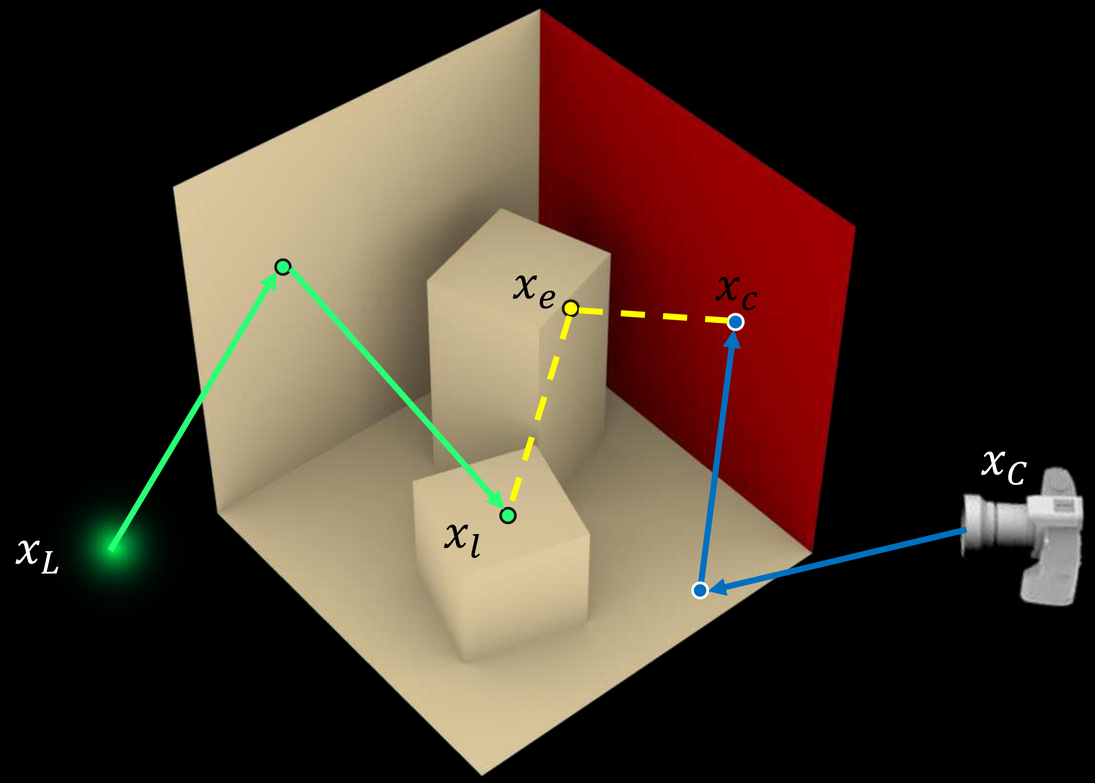
bidirectional path tracing (BDPT)

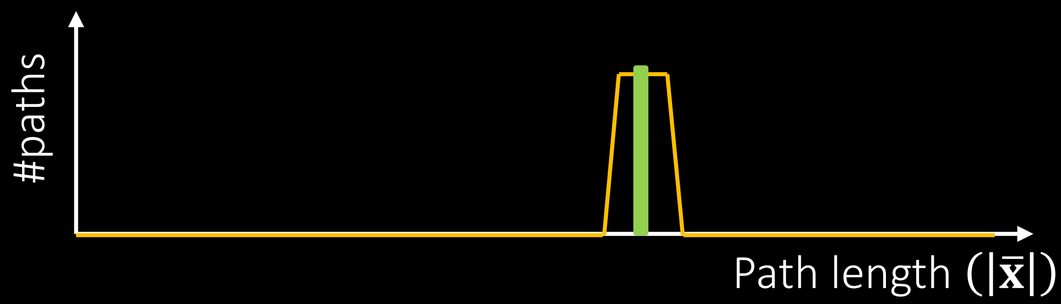
generate light sub-path, camera sub-path

join x_l and x_c via connecting vertex (x_e)

$$|x_e \rightarrow x_l| + |x_e \rightarrow x_c| = |\bar{\mathbf{x}}| - \underset{\text{sub-path lengths}}{\text{source + sensor}}$$

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

step 2: generate path with target length $|\bar{\mathbf{x}}|$

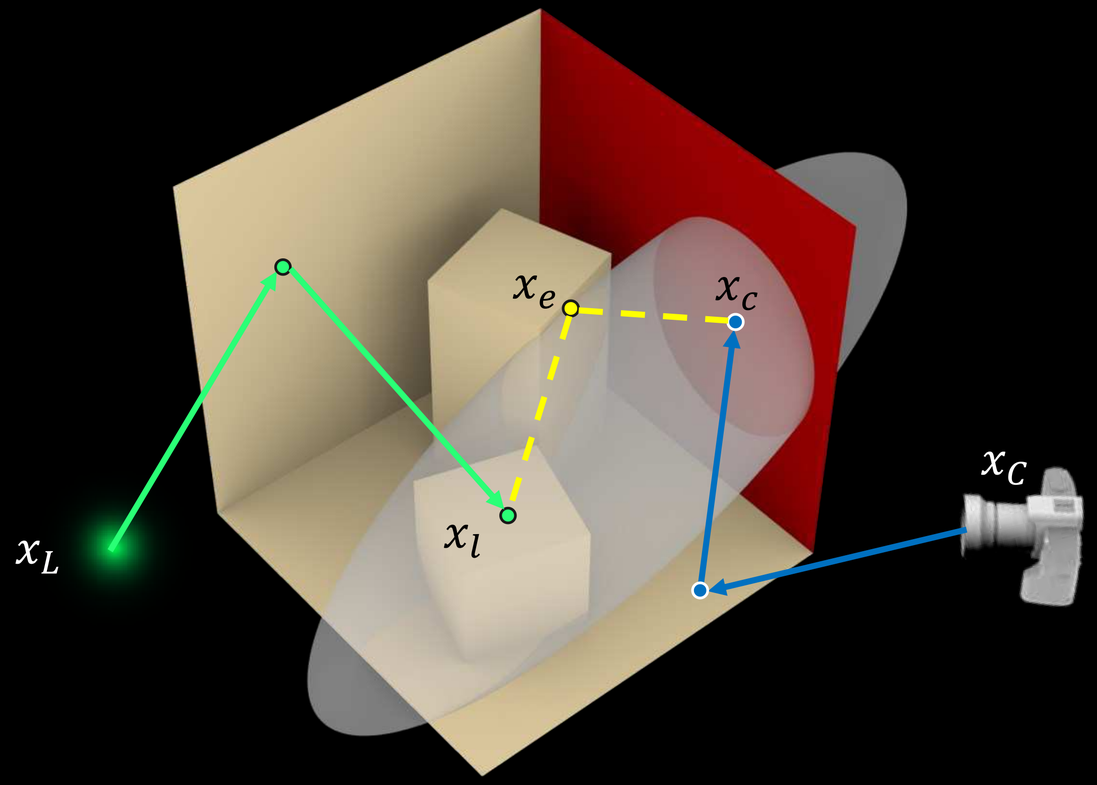
bidirectional path tracing (BDPT)

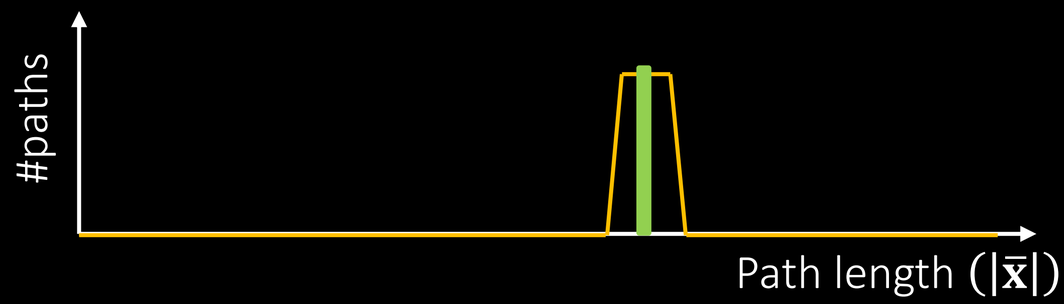
generate light sub-path, camera sub-path

join x_l and x_c via connecting vertex (x_e)

Definition of an ellipsoid: $|x_e \rightarrow x_l| + |x_e \rightarrow x_c| = |\bar{\mathbf{x}}| -$ source + sensor sub–path lengths

Path sampling for time-gated case





step 1: sample path length $|\bar{\mathbf{x}}|$

step 2: generate path with target length $|\bar{\mathbf{x}}|$

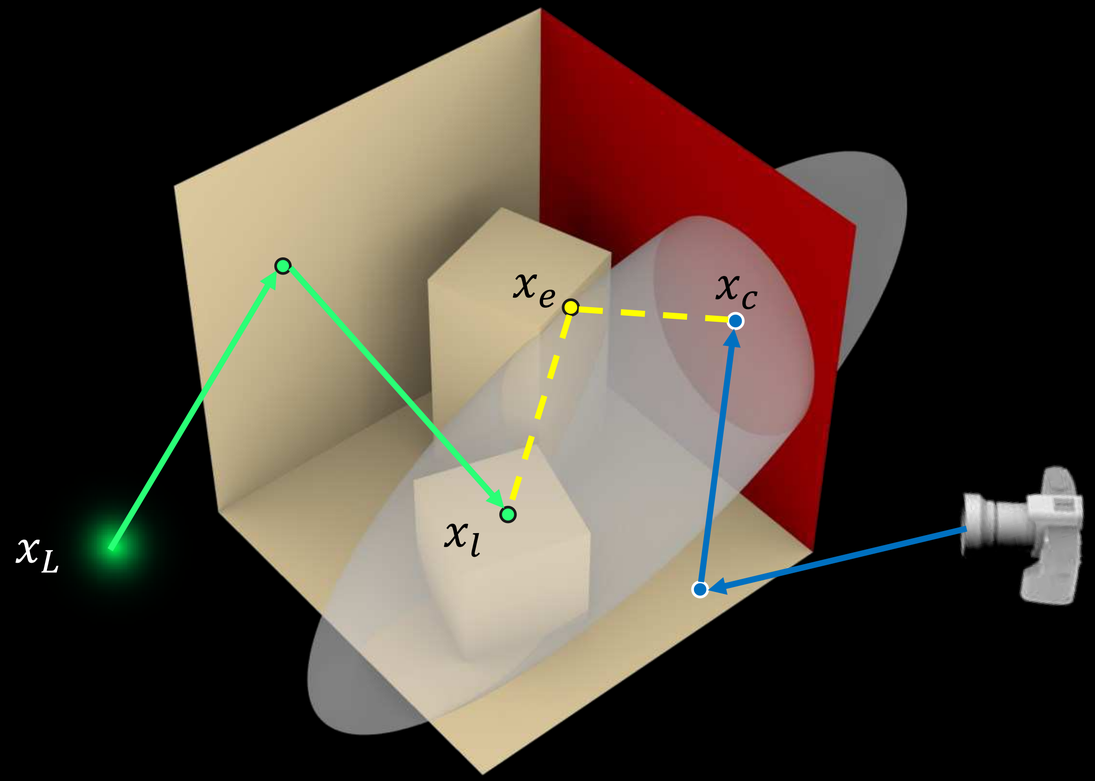
bidirectional path tracing (BDPT)

generate light sub-path, camera sub-path

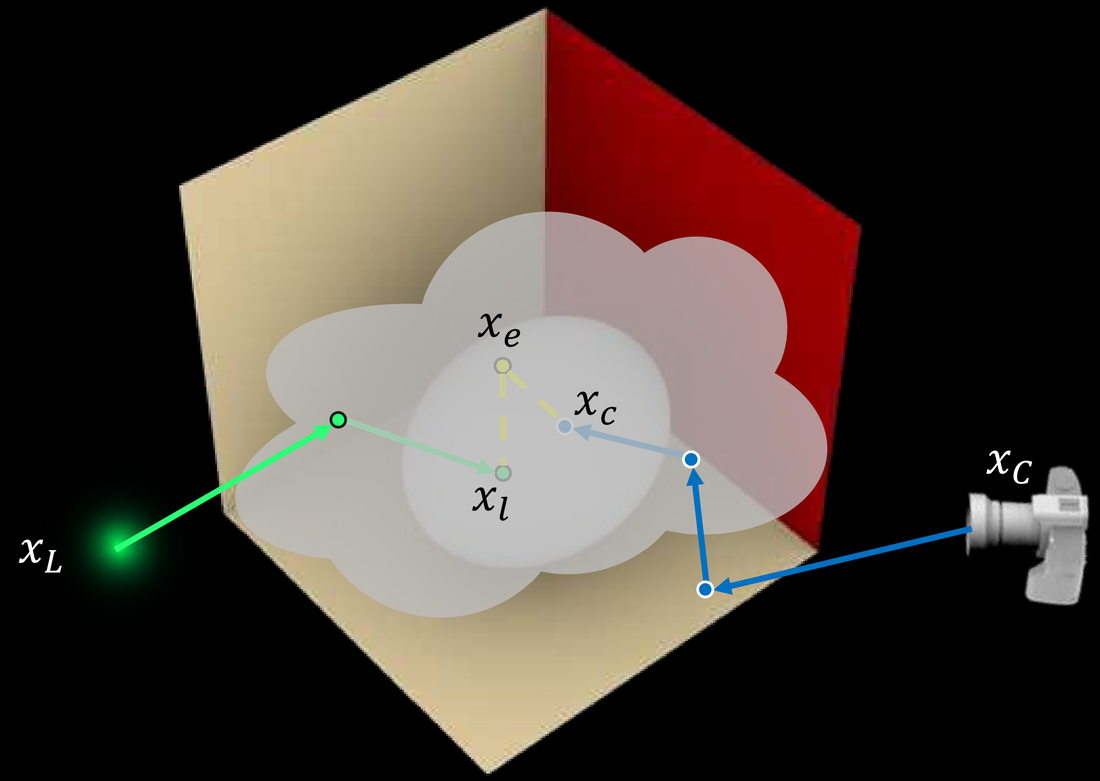
join x_l and x_c via connecting vertex (x_e)

Definition of an ellipsoid: $|x_e \rightarrow x_l| + |x_e \rightarrow x_c| = |\bar{\mathbf{x}}| -$ source + sensor sub–path lengths

Path sampling for time-gated case

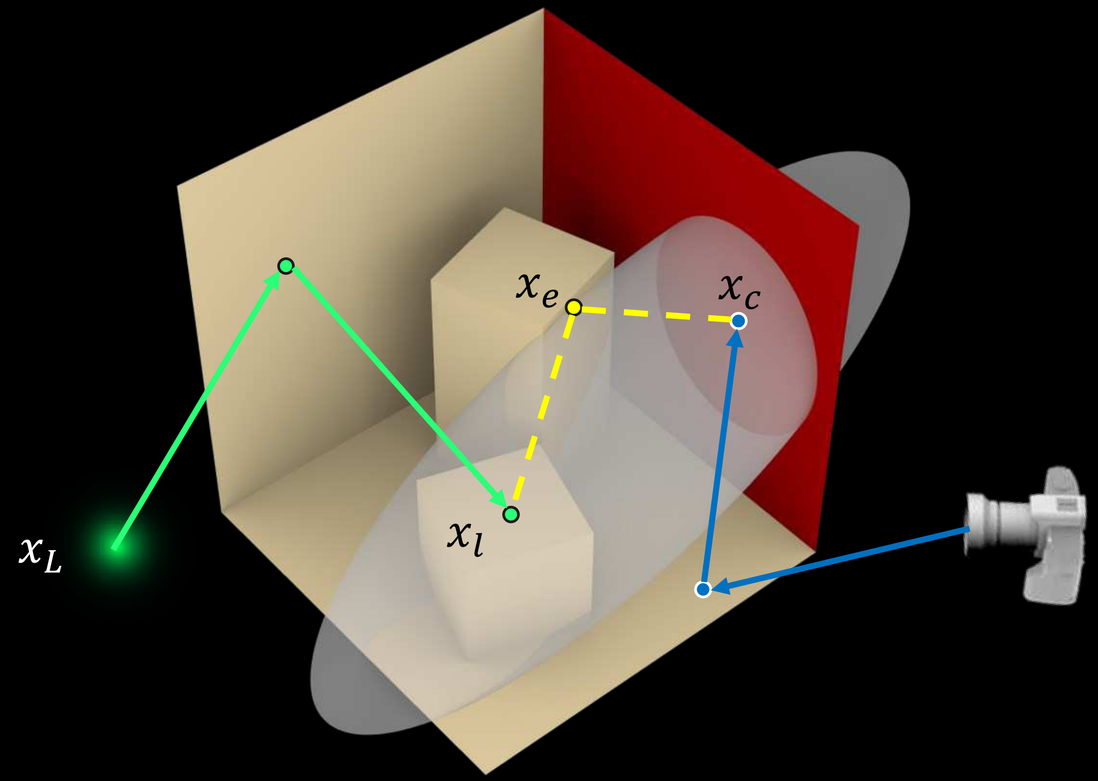


surface case

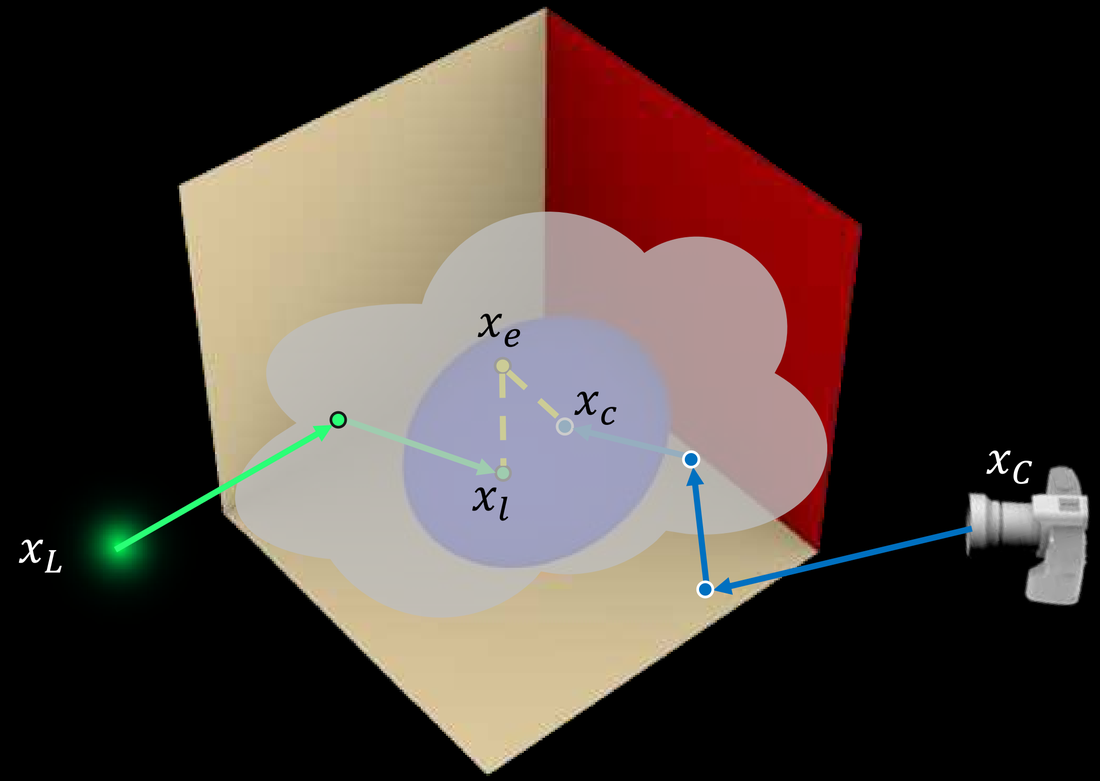


volumetric case

Path sampling for time-gated case

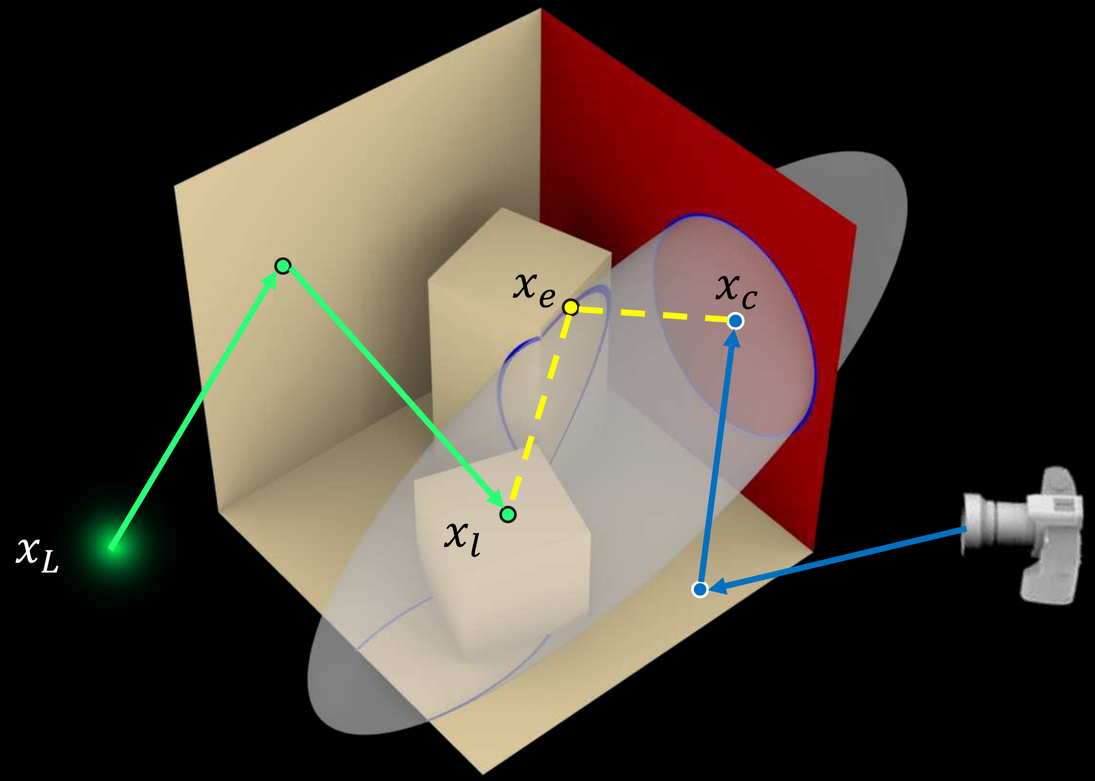


surface case

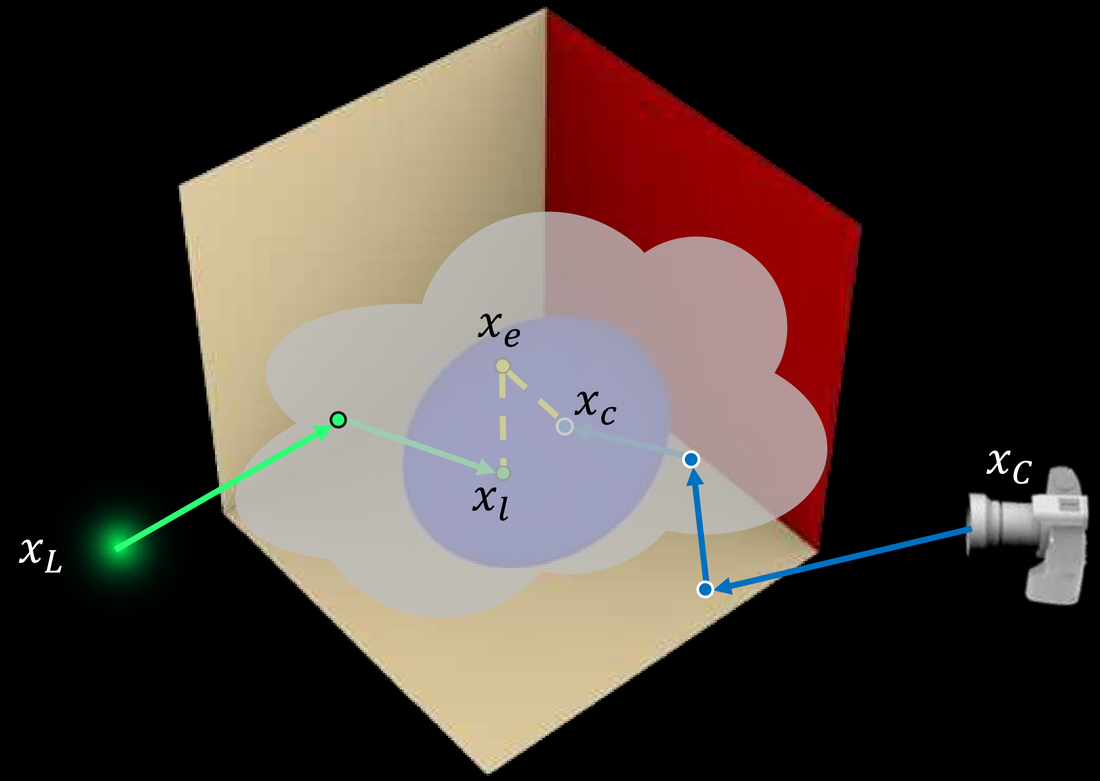


volumetric case

Path sampling for time-gated case

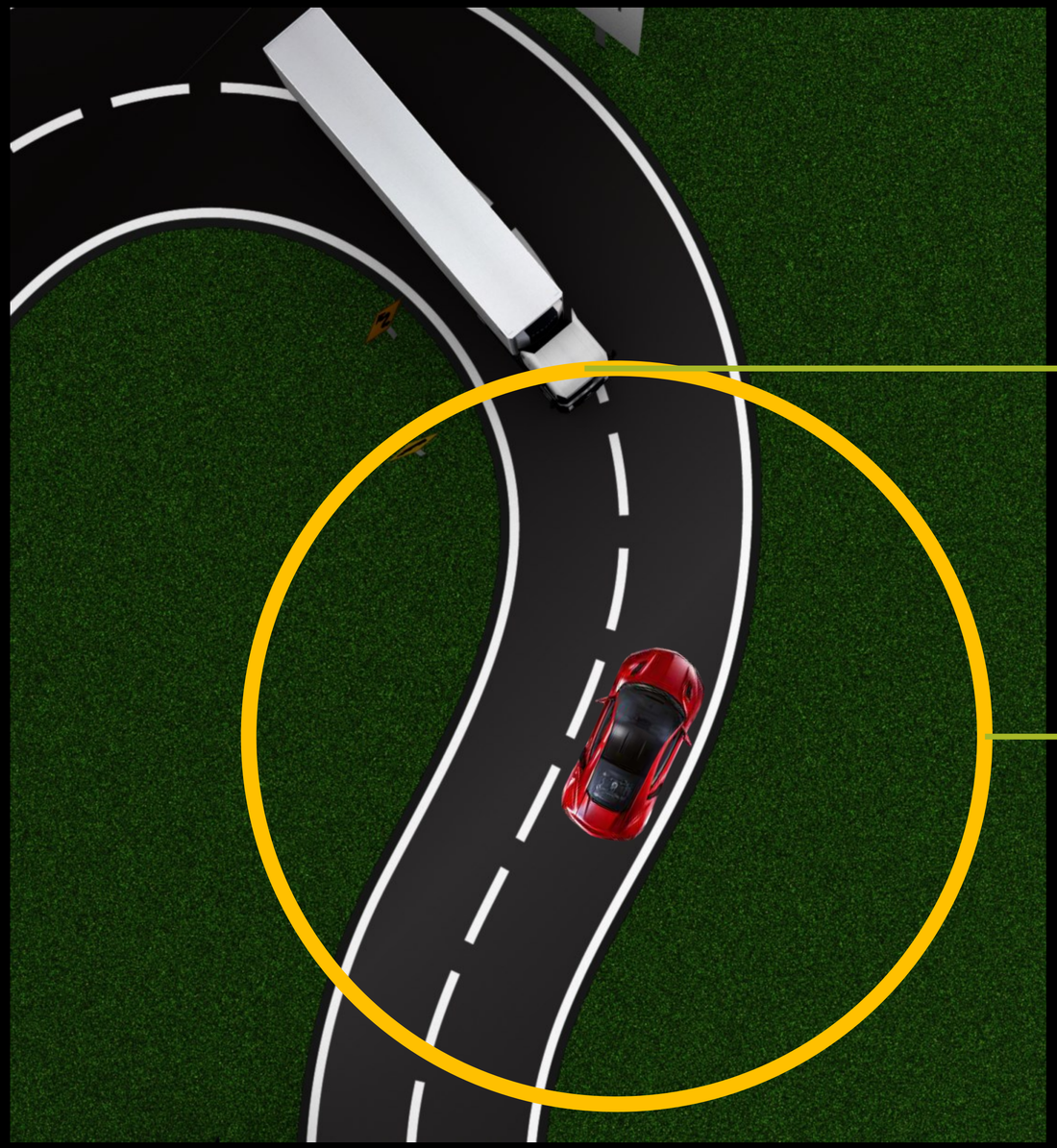


surface case



volumetric case

Application: proximity detector for cars



proximity detected

virtual light curtain

Application: proximity detector for cars

road scene

standard BDPT

BDPT w/ ellipsoidal connections

Gate width: 200 ps (1.14% scene)
Rendering time: 10s per frame

Application: proximity detector for cars

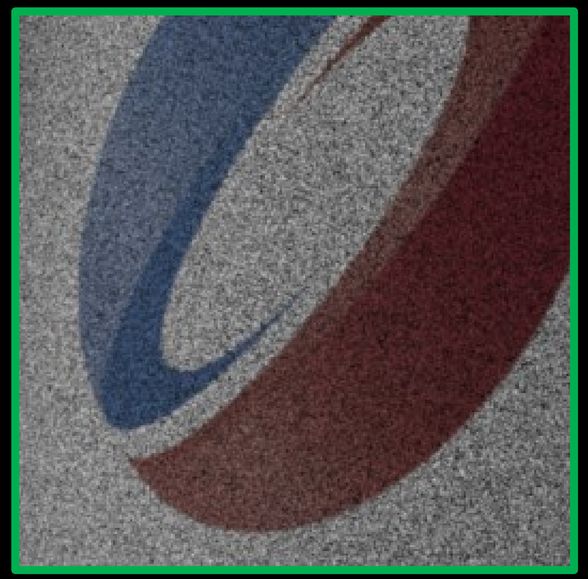
road scene

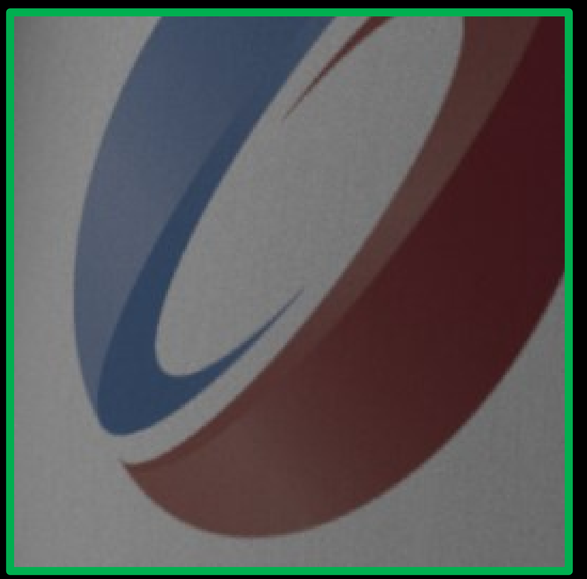
standard BDPT

BDPT w/ ellipsoidal connections

time = 1.34s

time = 1.74s



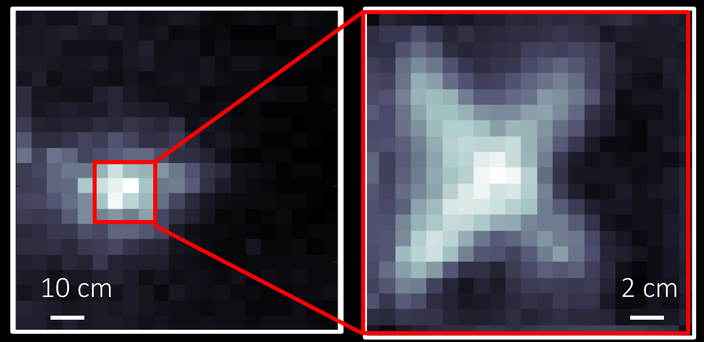


Imaging projects using this renderer

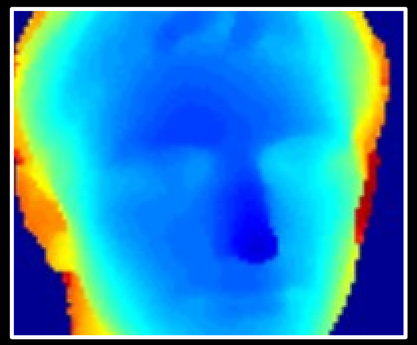
Mitsuba based
open source implementation

http://bit.ly/32Uzm1m

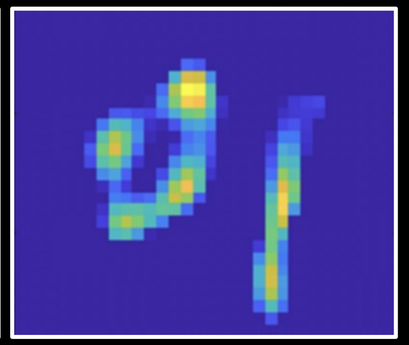
non-line-of-sight



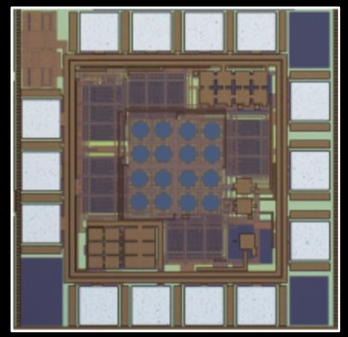
depth sensing



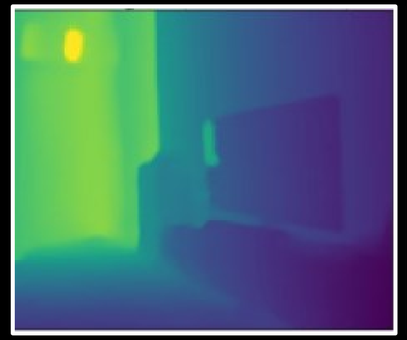
tissue imaging

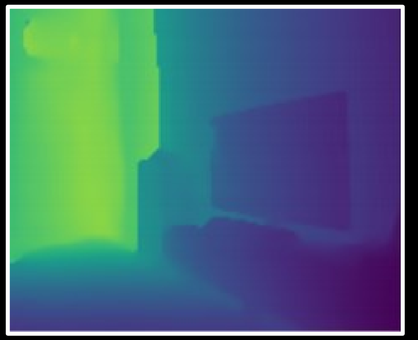


sensor design [White et al. 2022]]

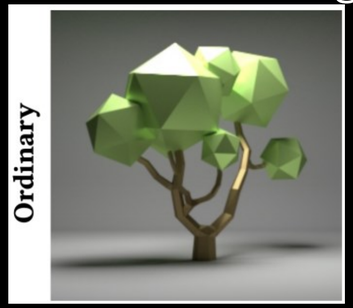


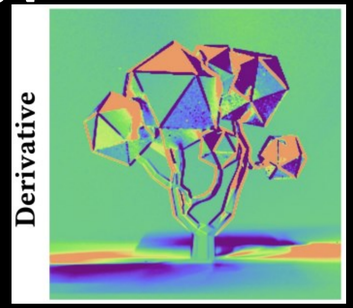
deep learning [Barragan et al. 2021]



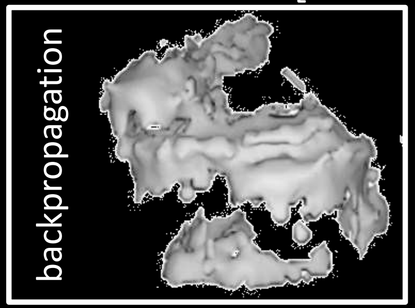


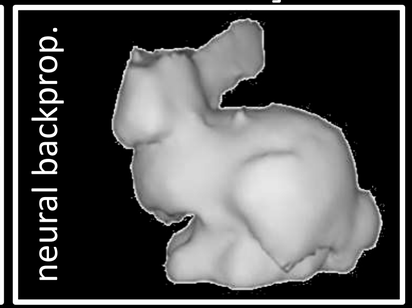
inverse rendering [Wu et al. 2021]





SONAR [Reed et al. 2023]



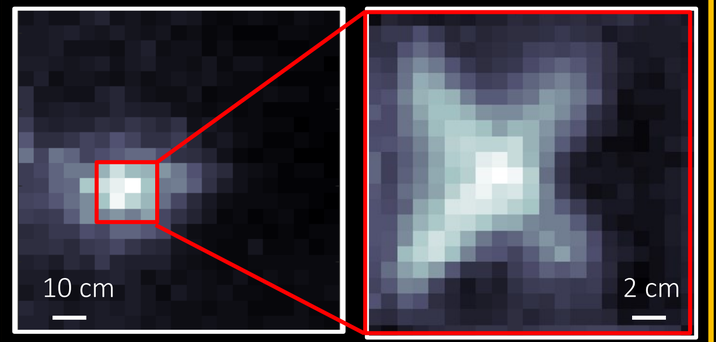


Imaging projects using this renderer

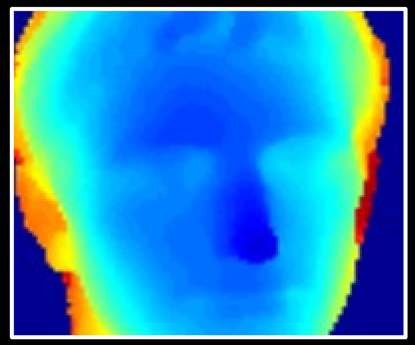
Mitsuba based
open source implementation

http://bit.ly/32Uzm1m

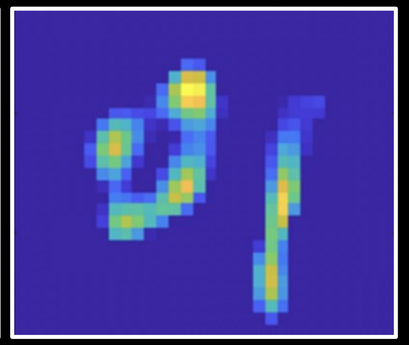
non-line-of-sight



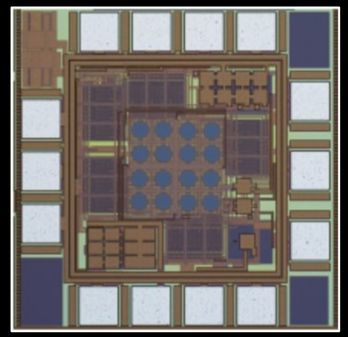
depth sensing



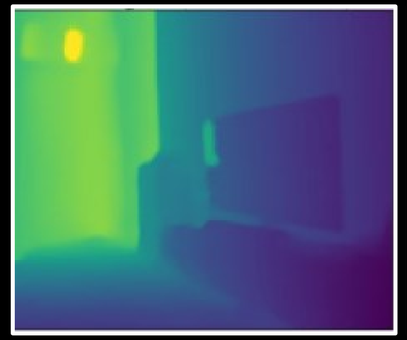
tissue imaging

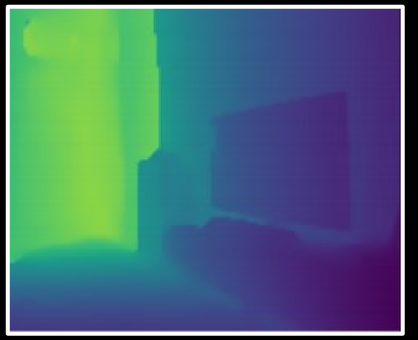


sensor design [White et al. 2022]]

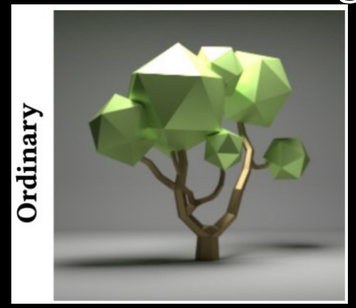


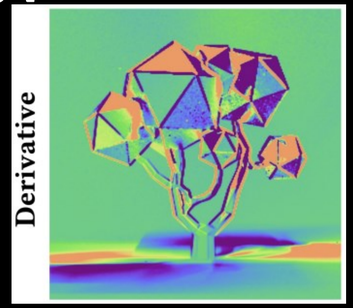
deep learning [Barragan et al. 2021]



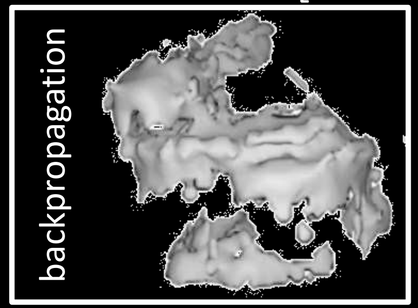


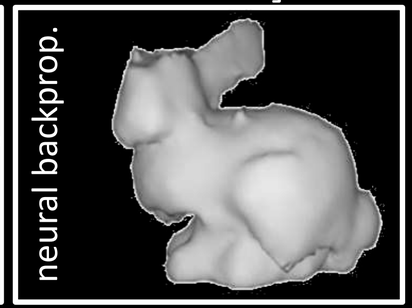
inverse rendering [Wu et al. 2021]





SONAR [Reed et al. 2023]



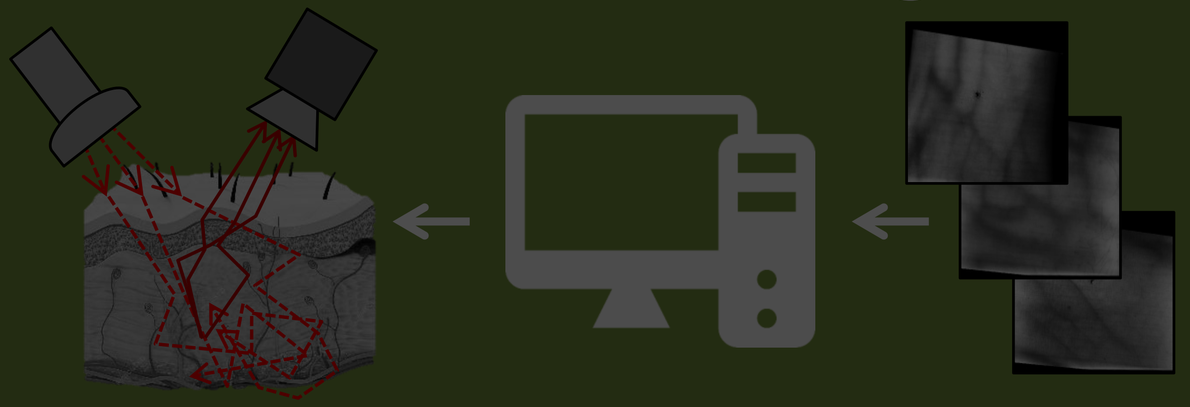


Physics-based rendering and its applications to computational imaging

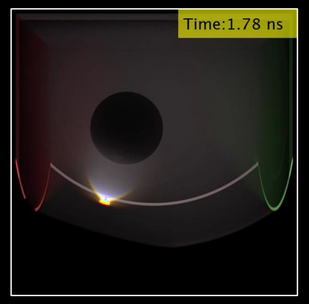
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

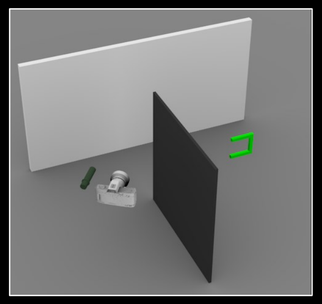
inverse rendering



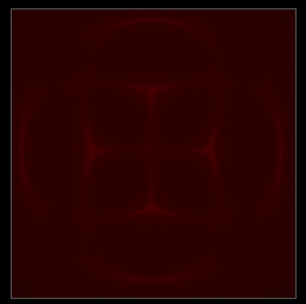
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



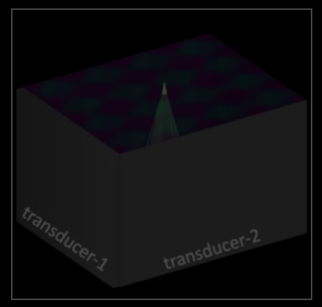
time-of-flight imaging



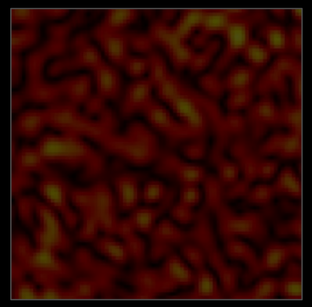
non-line-of-sight imaging



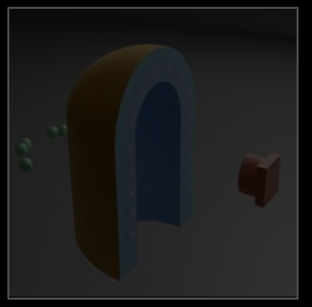
acousto-optic lensing



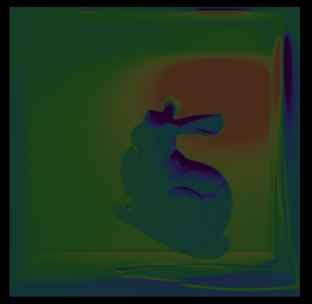
ultrafast light scanning



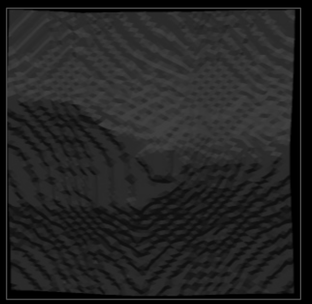
speckle imaging



tactile sensor design

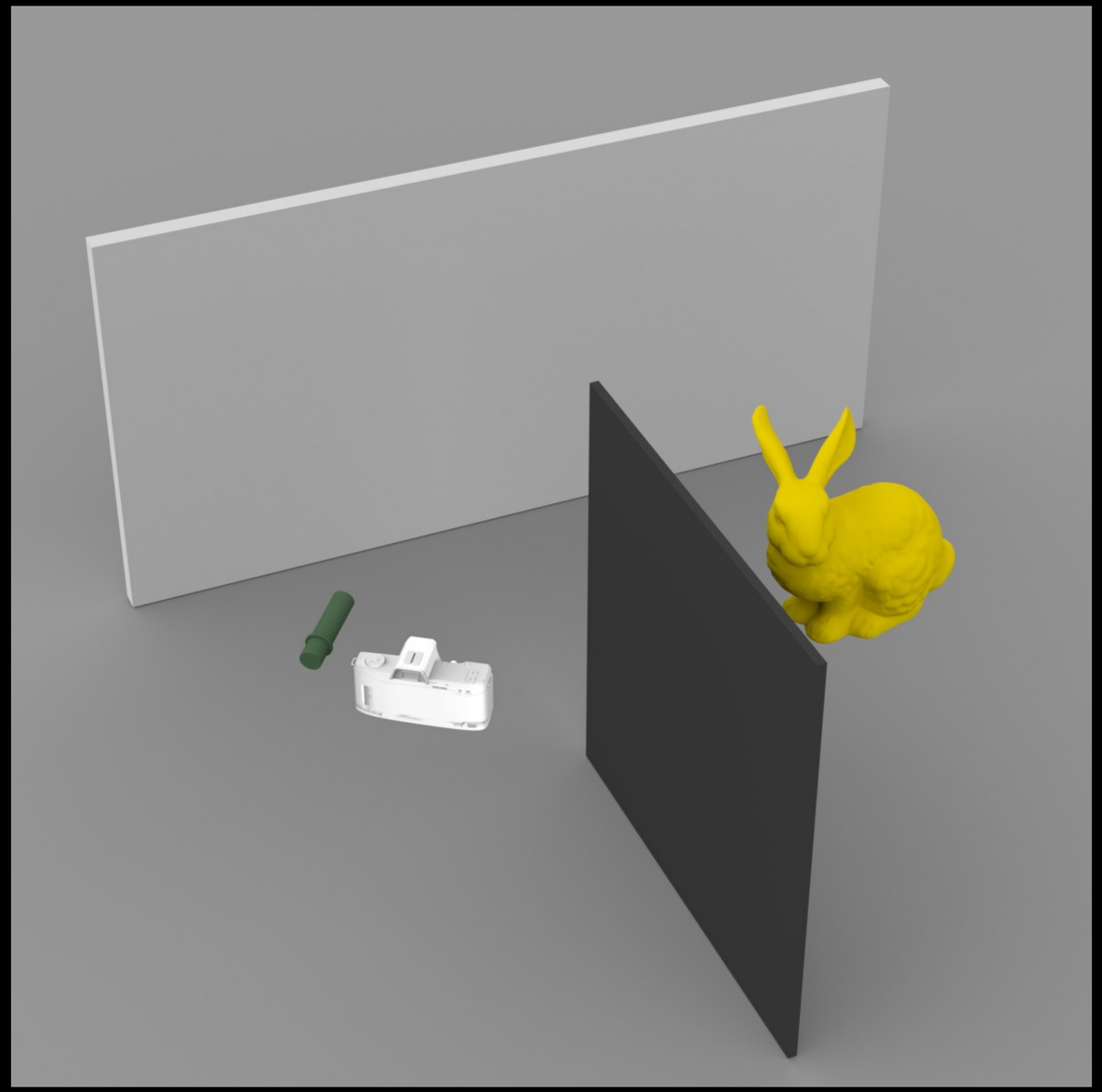


differentiable renderer

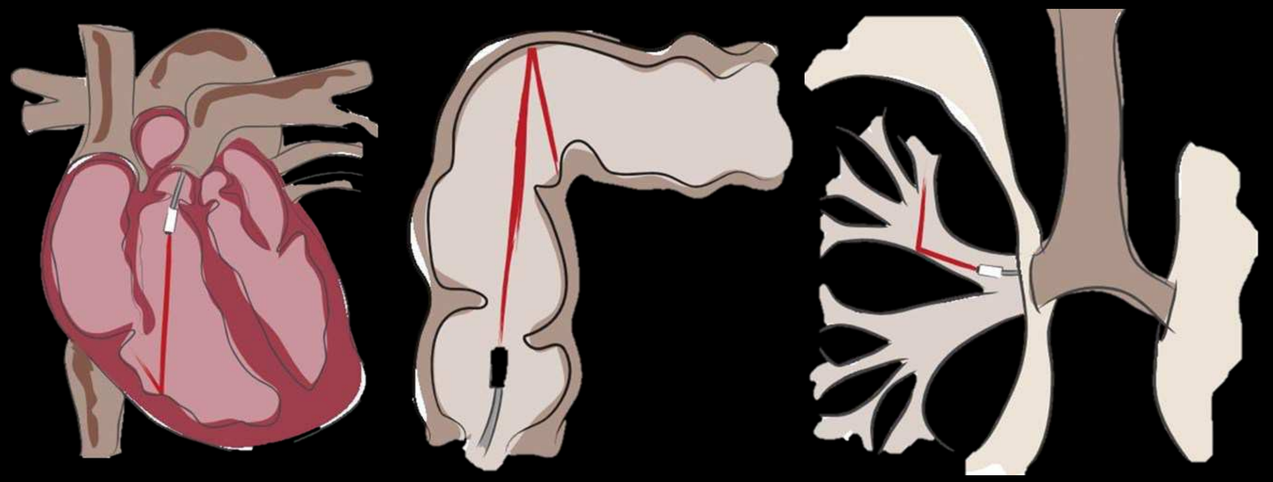


inverse problems

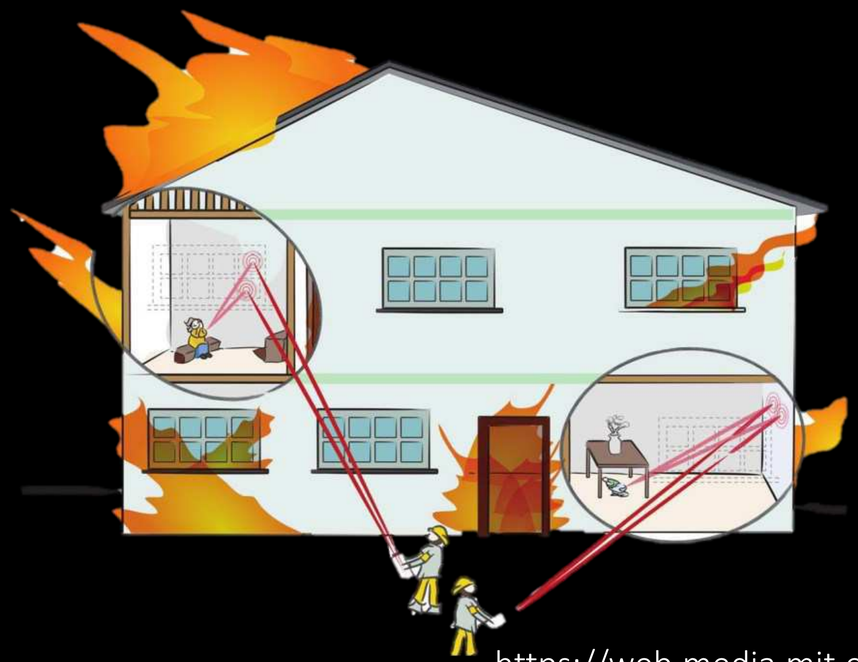
Non-line-of-sight imaging



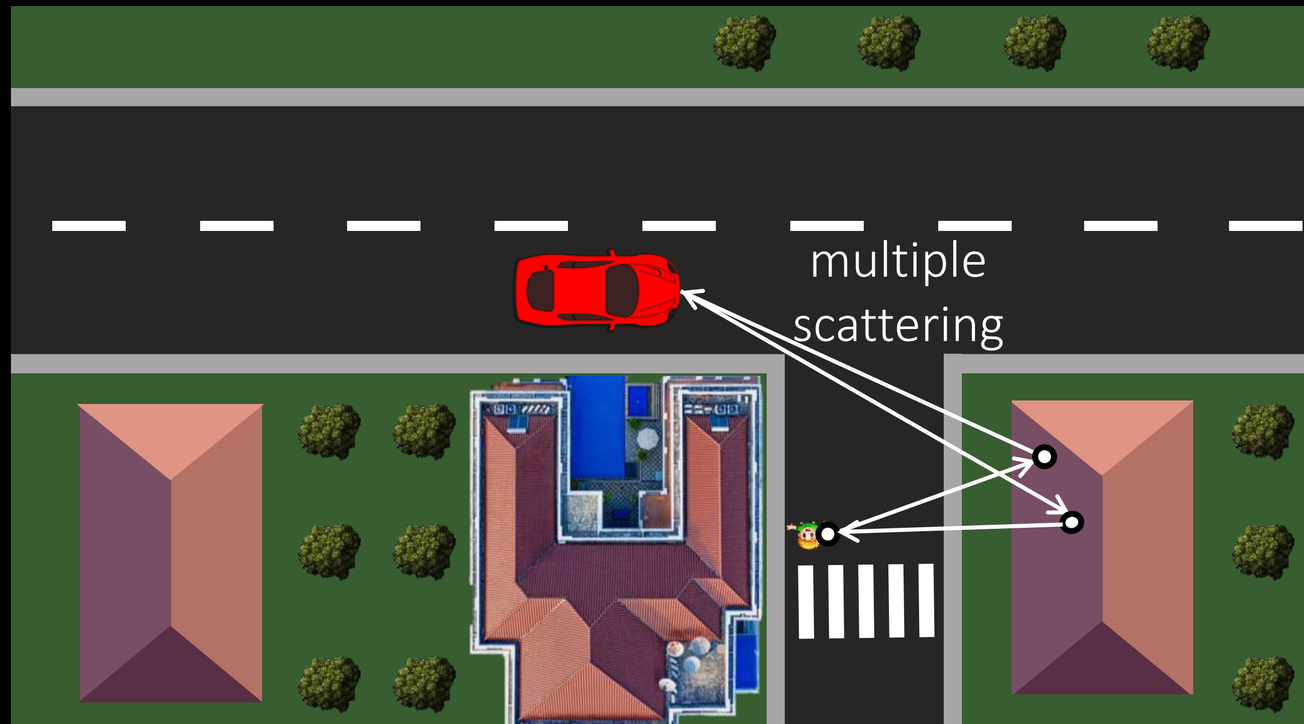
Applications



https://web.media.mit.edu/~raskar/cornar/

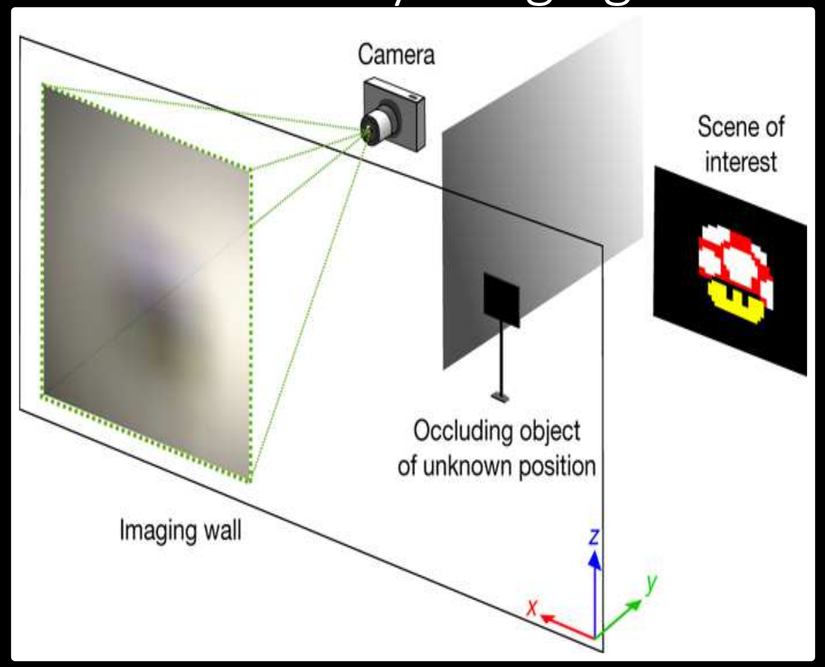


https://web.media.mit.edu/~raskar/cornar/



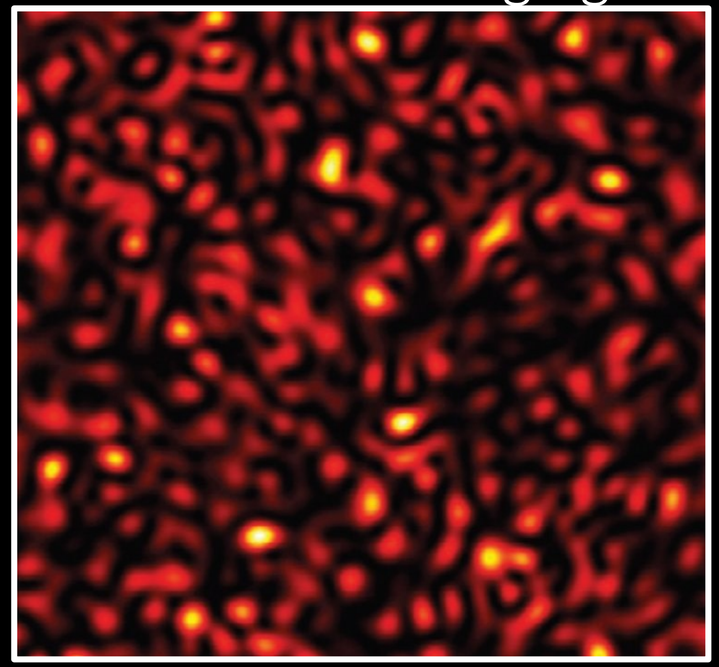
A lot of research on non-line-of-sight imaging

intensity imaging



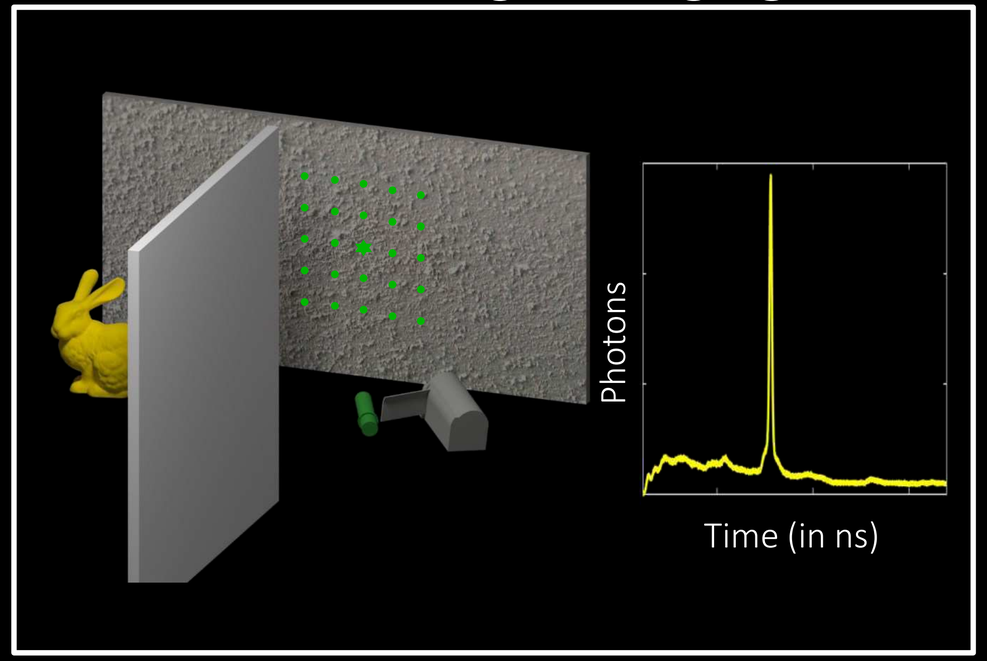
Bouman et al., ICCV 2017
Saunders et al., Nature 2019
Saunders et al., COSI 2019
Maeda et al., ICCP 2019
Lin et al., COSI 2020
Sharma, ICCV 2021

coherence imaging



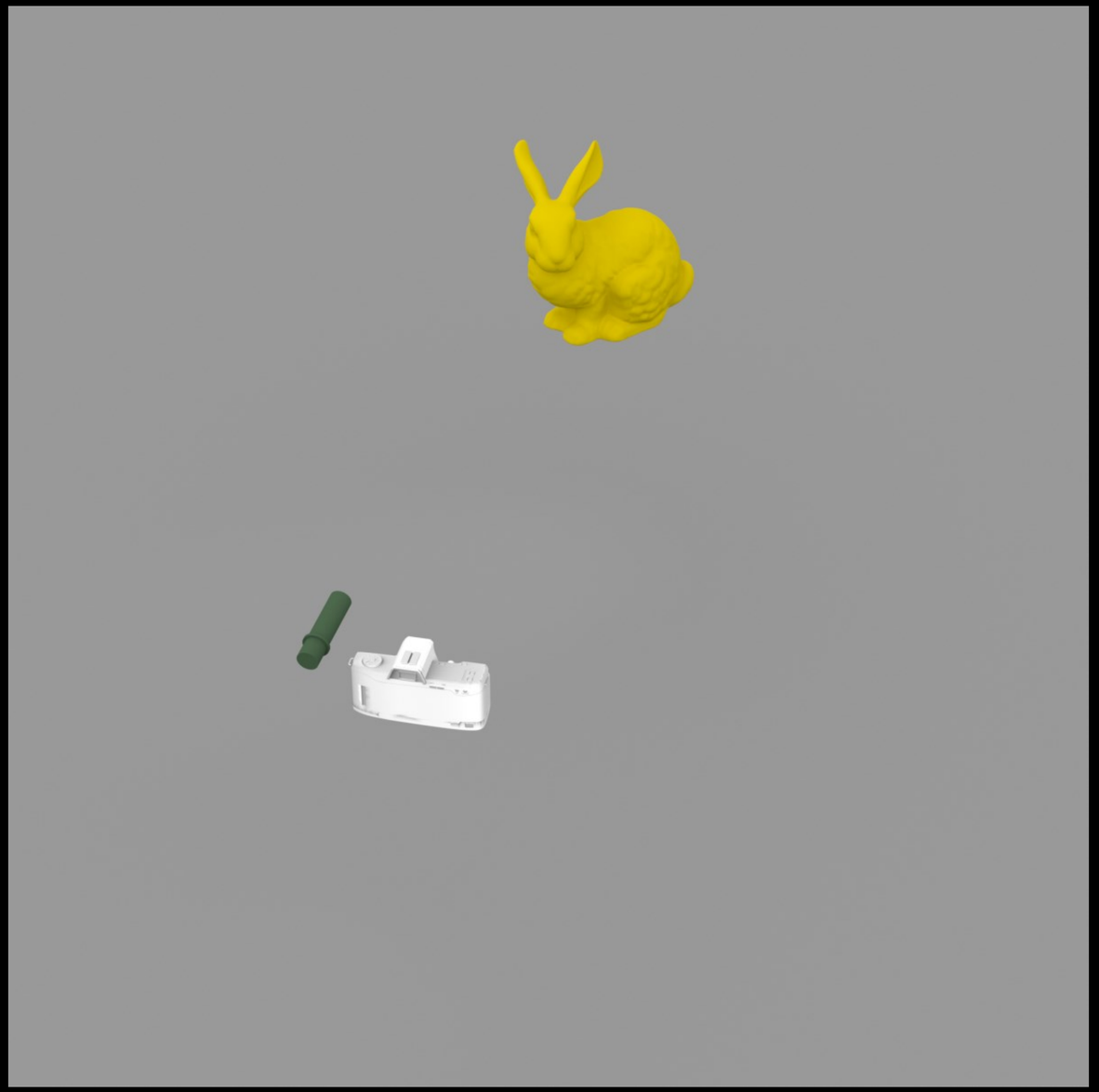
Katz et al., Nat. Photonics 2014
Lei et al., CVPR, 2019
Boger-Lombard, Nat. Com. 2019
Metzler et al., Optica 2020
Willomitzer et al., Nat. Com. 2021
Chen et al. SPIE 2022

time-of-flight imaging



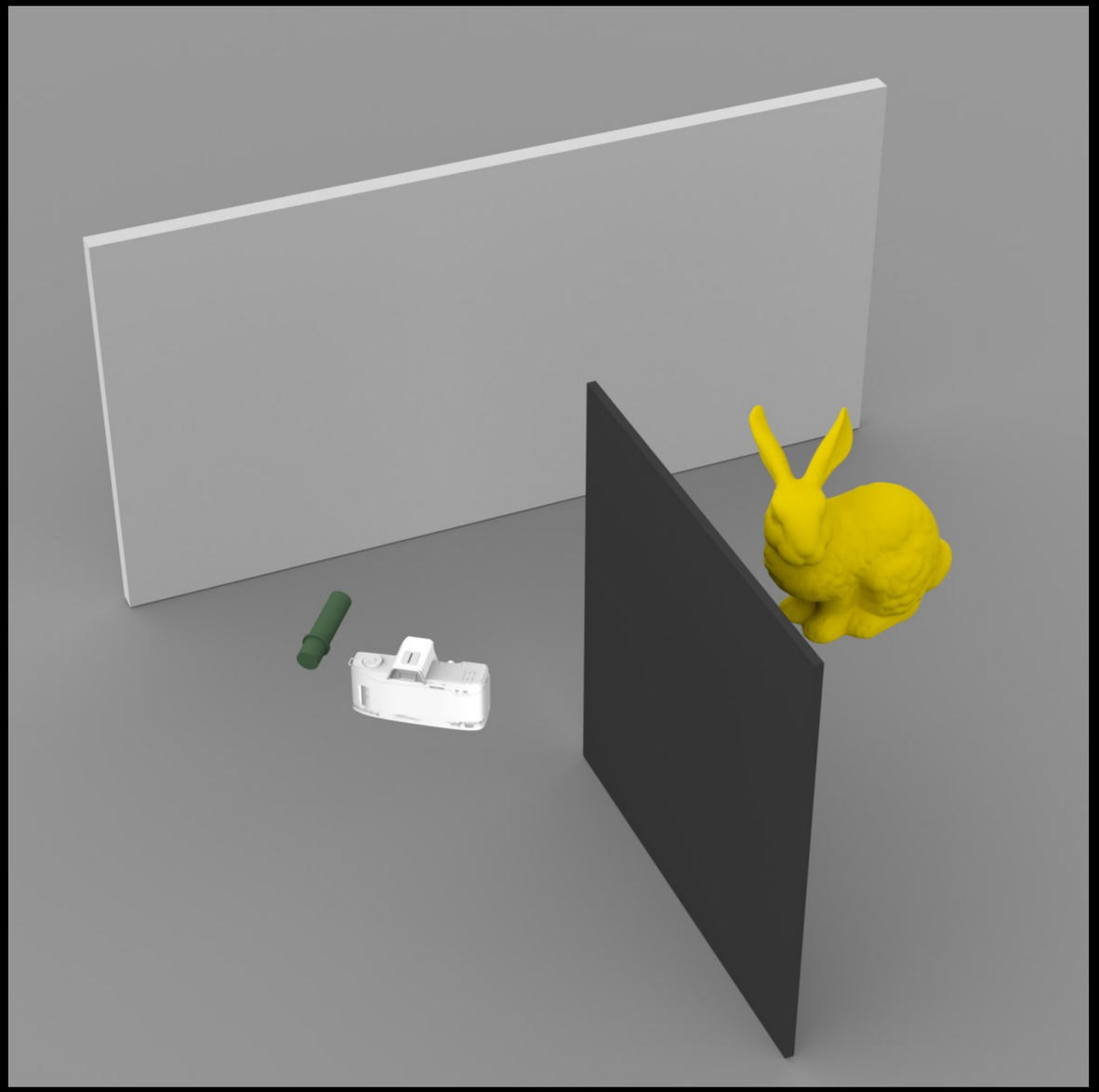
Velten et al., Nat. comm. 2012
Toole et al., Nature 2018
Liu et al., Nature 2019, Nat. comm. 2020
Rapp et al. Nat. comm. 2020
Xin et al., CVPR 2019
Nam et al., Nat. comm. 2021

How do we image in line-of-sight?

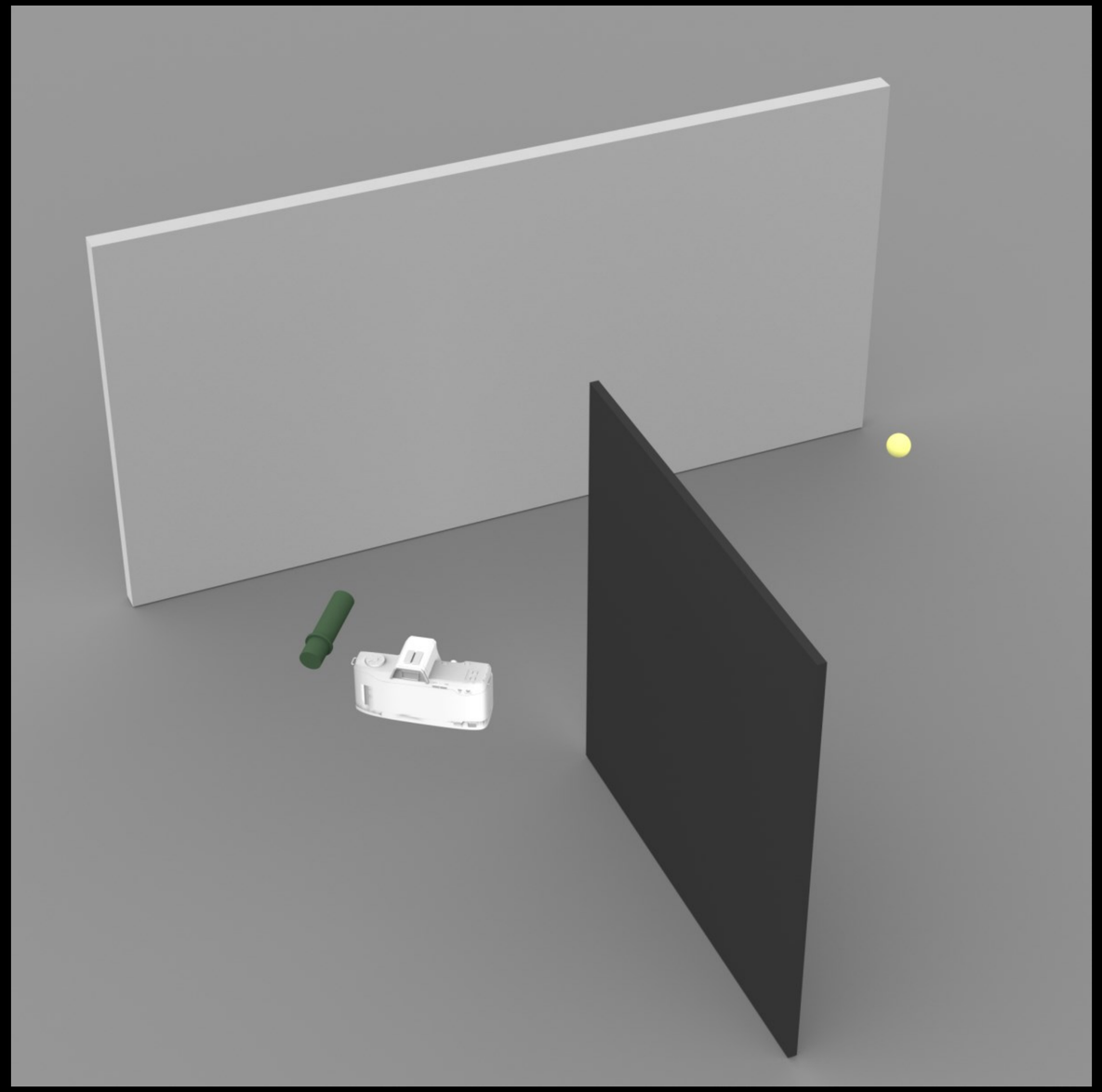


How do we image in line-of-sight?

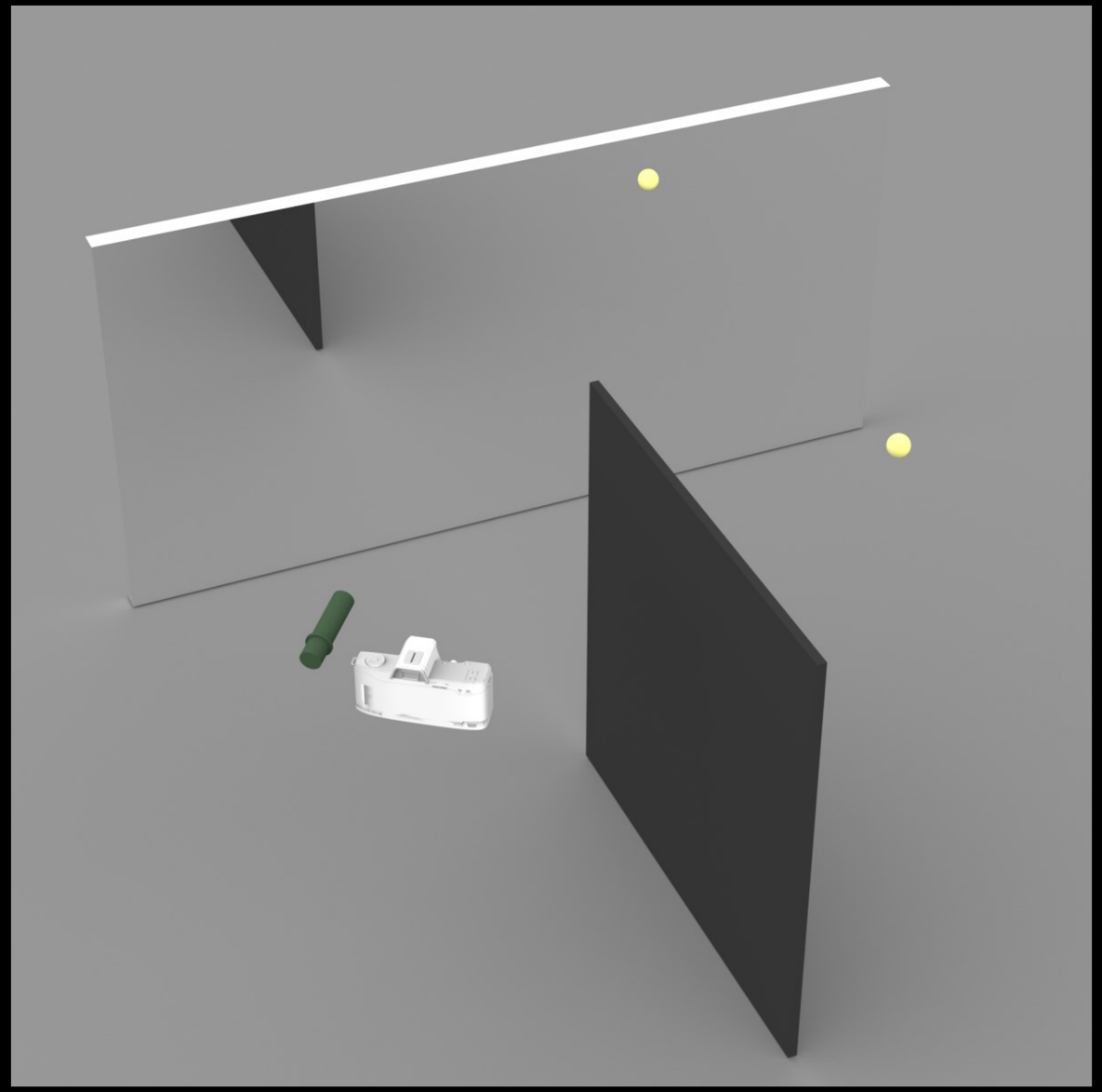
How do we image in non-line-of-sight?



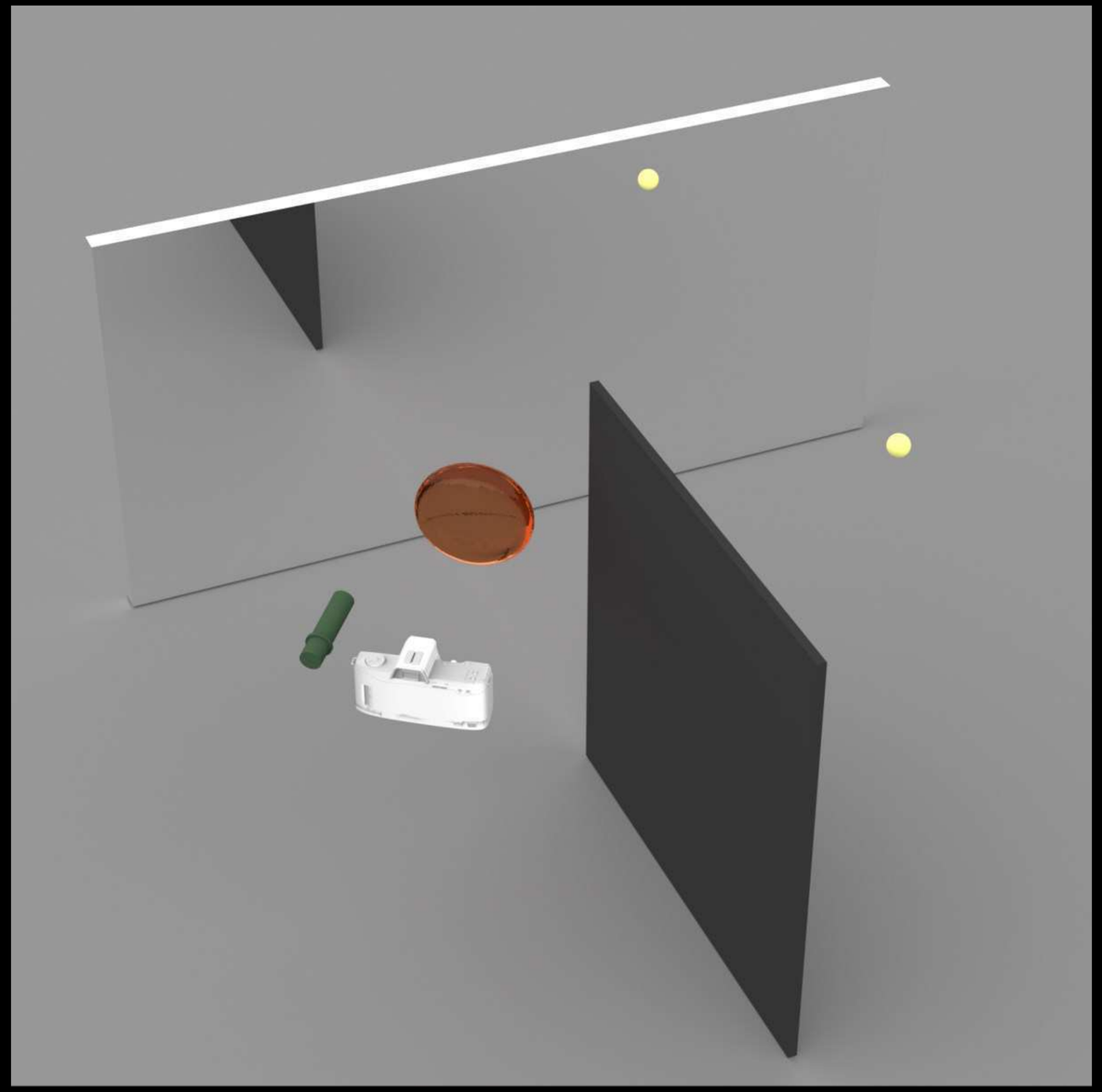
How do we focus on a voxel?



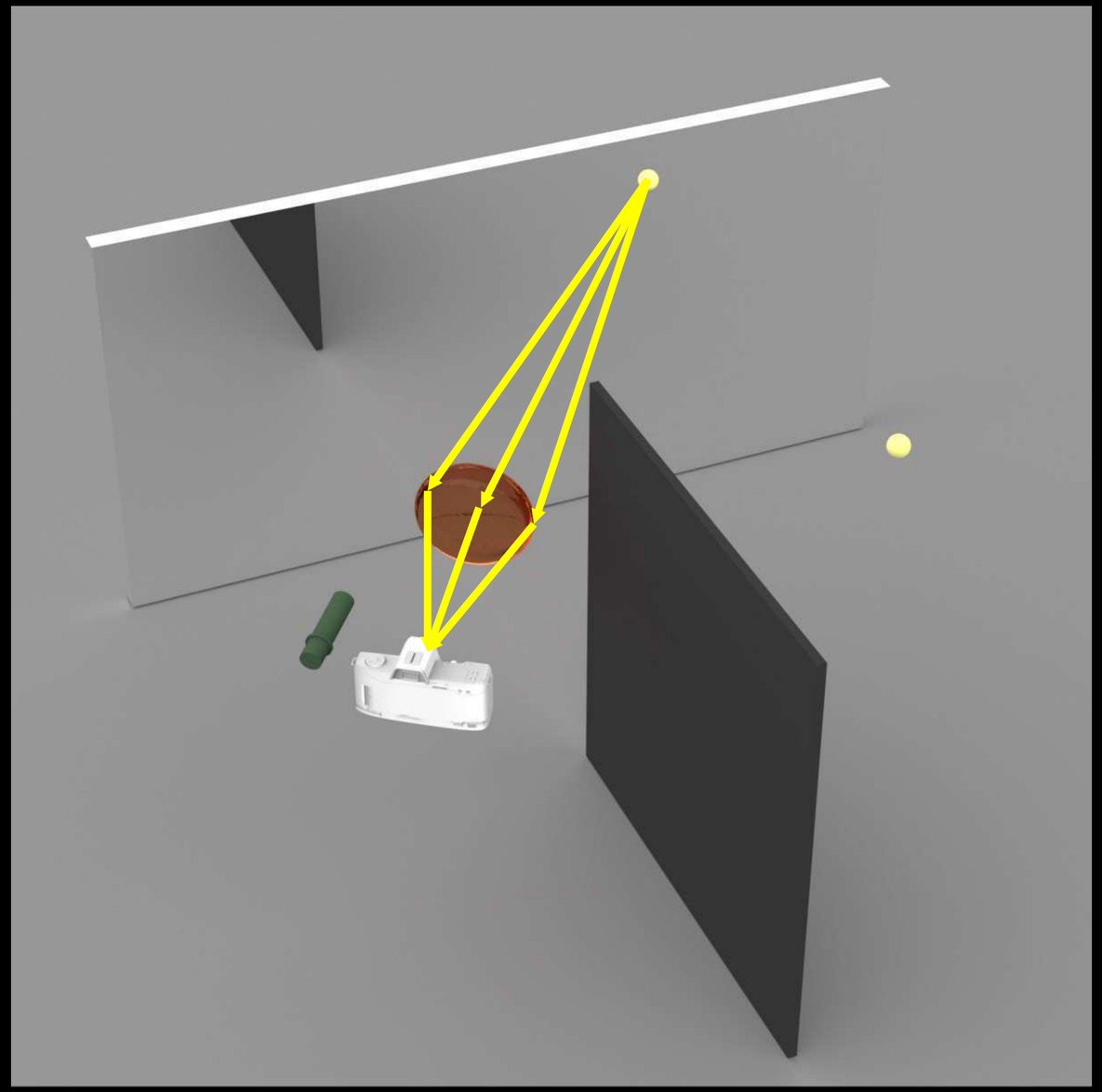
How do we focus on a voxel?



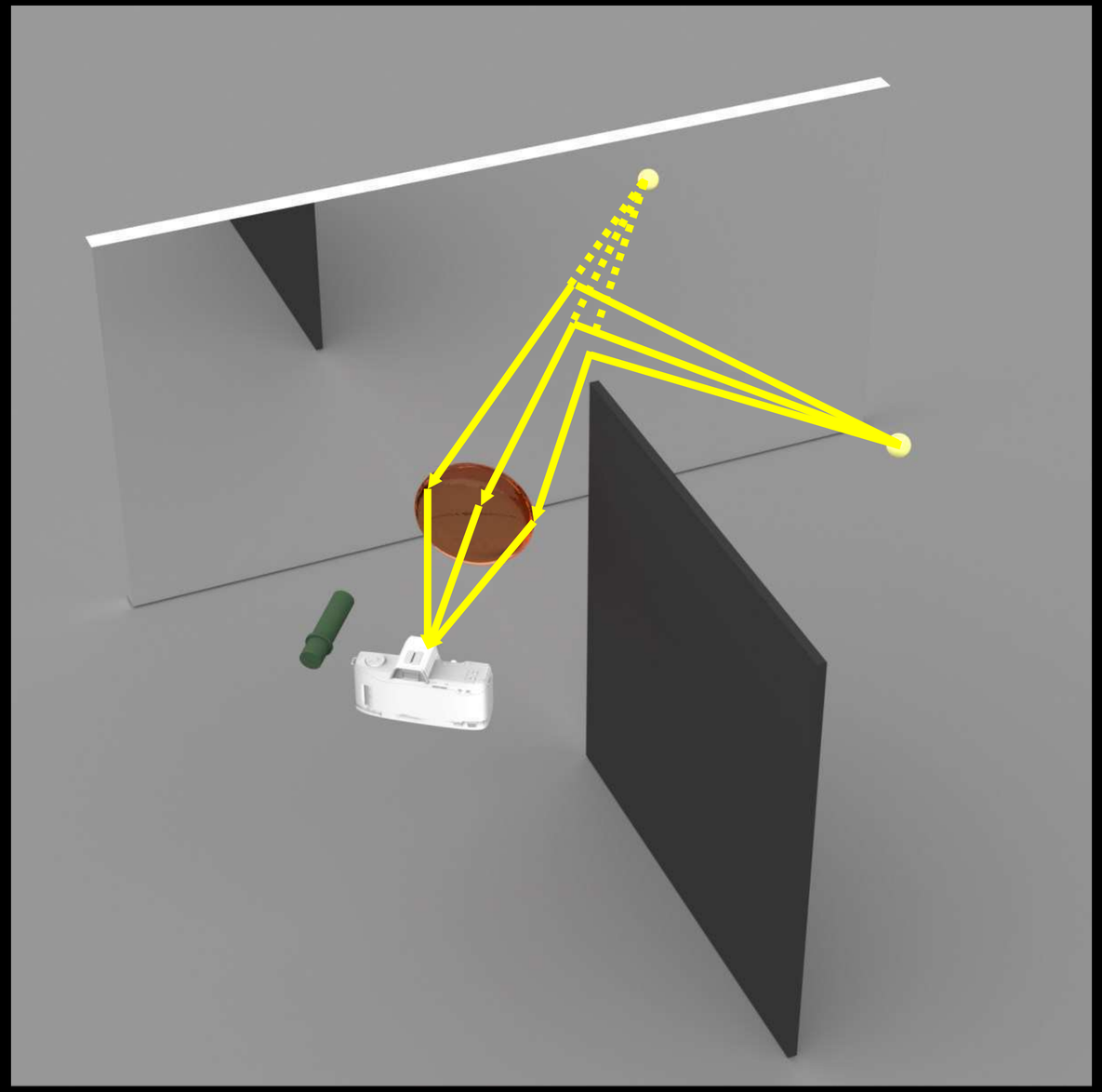
How do we focus on a voxel?



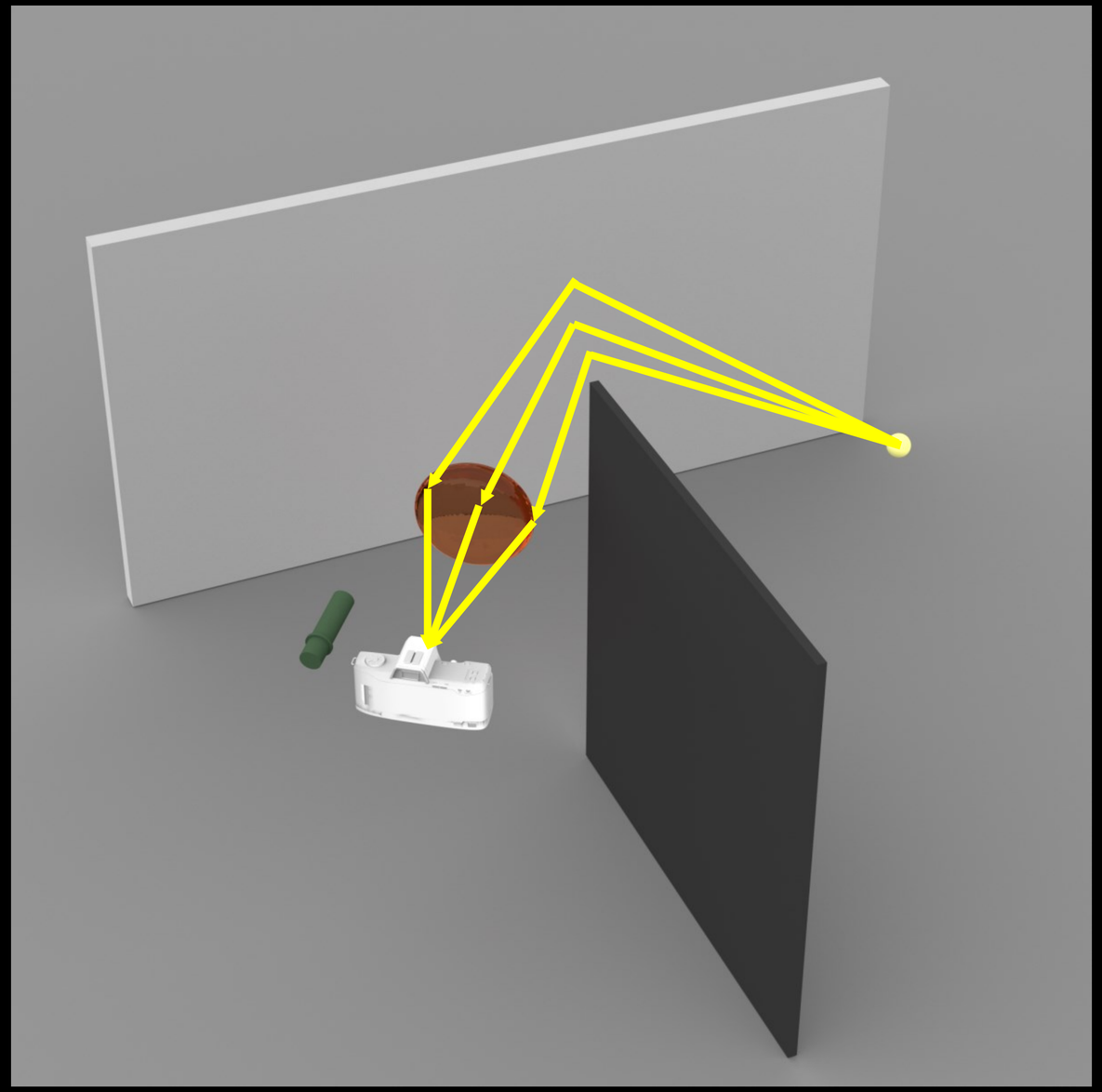
How do we focus on a voxel?



How do we focus on a voxel?



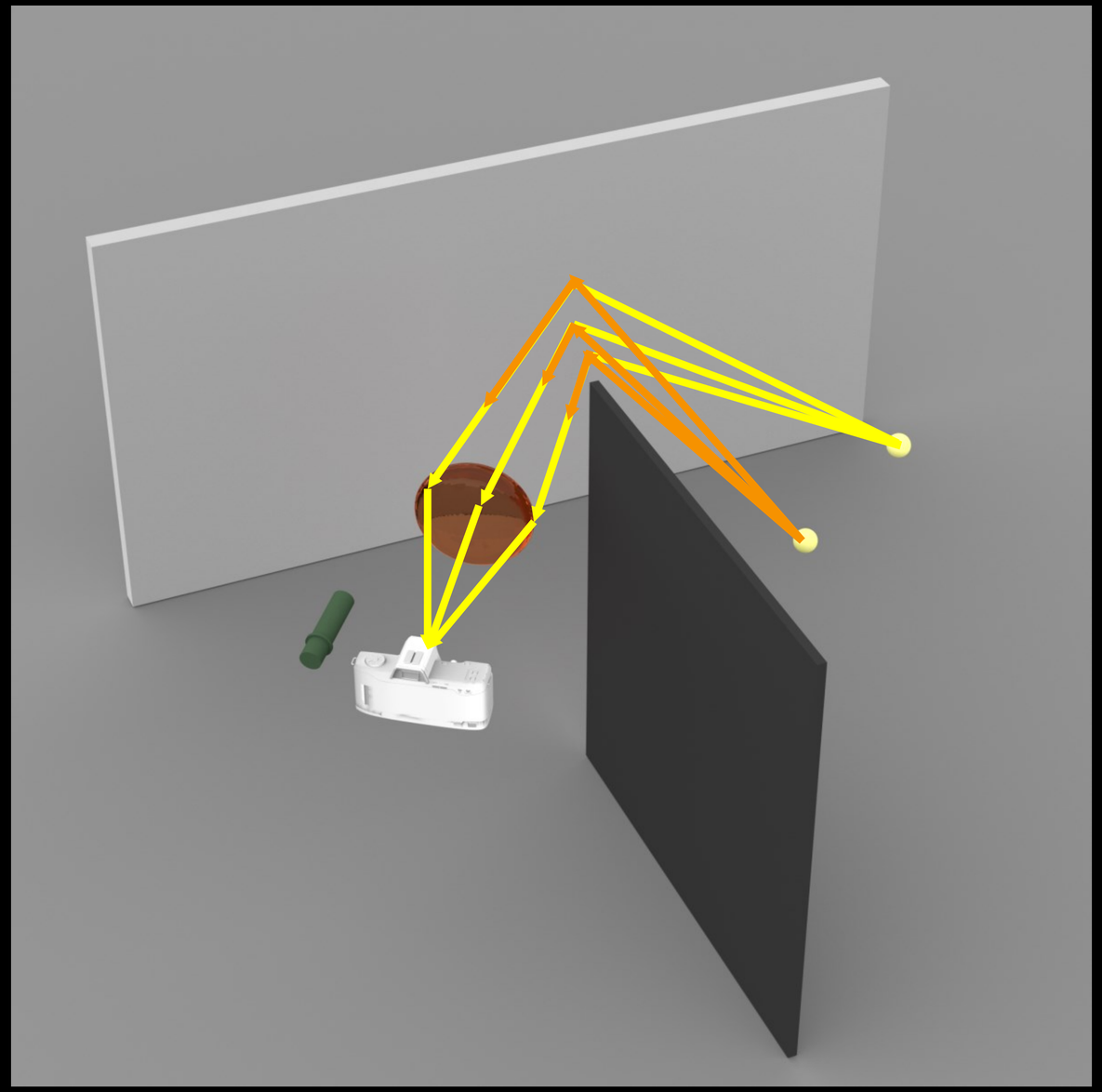
How do we focus on a voxel?



How do we focus on a voxel?

challenge:

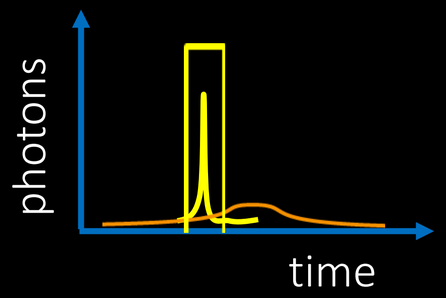
out of focus voxels



How do we focus on a voxel?

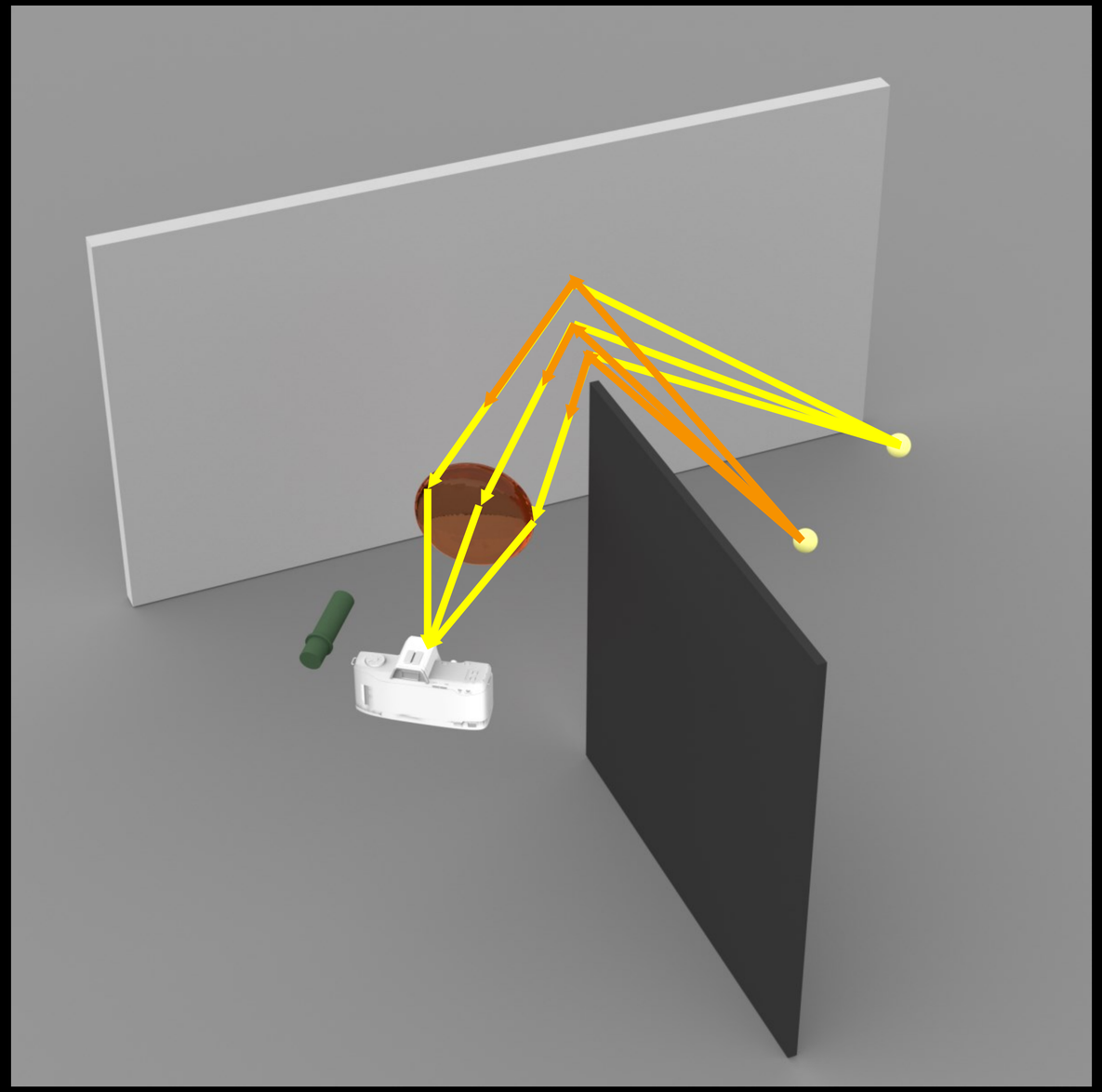
challenge:

out of focus voxels



solution:

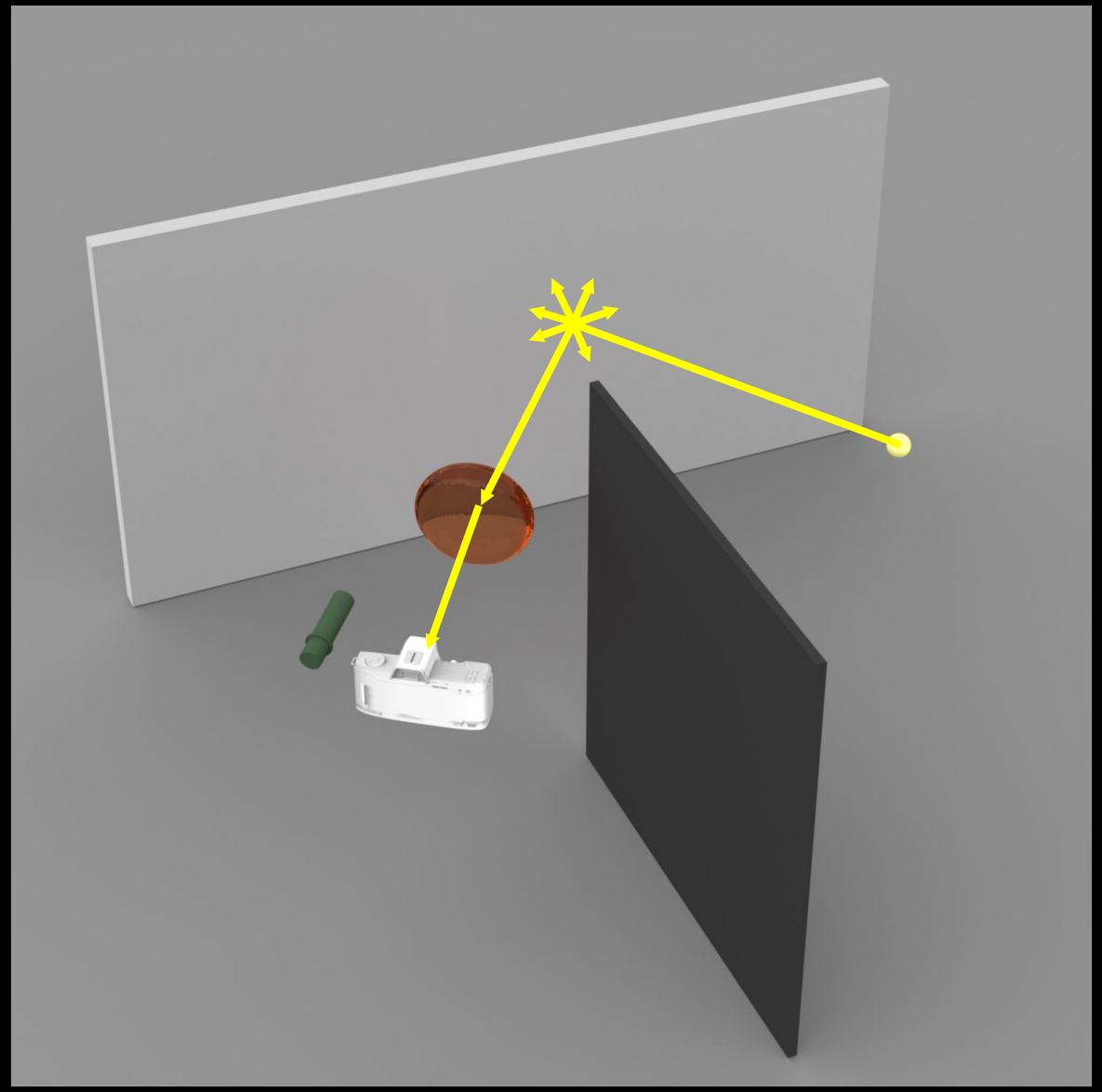
time-gate photons



How do we focus on a voxel?

challenge:

non-specular photons



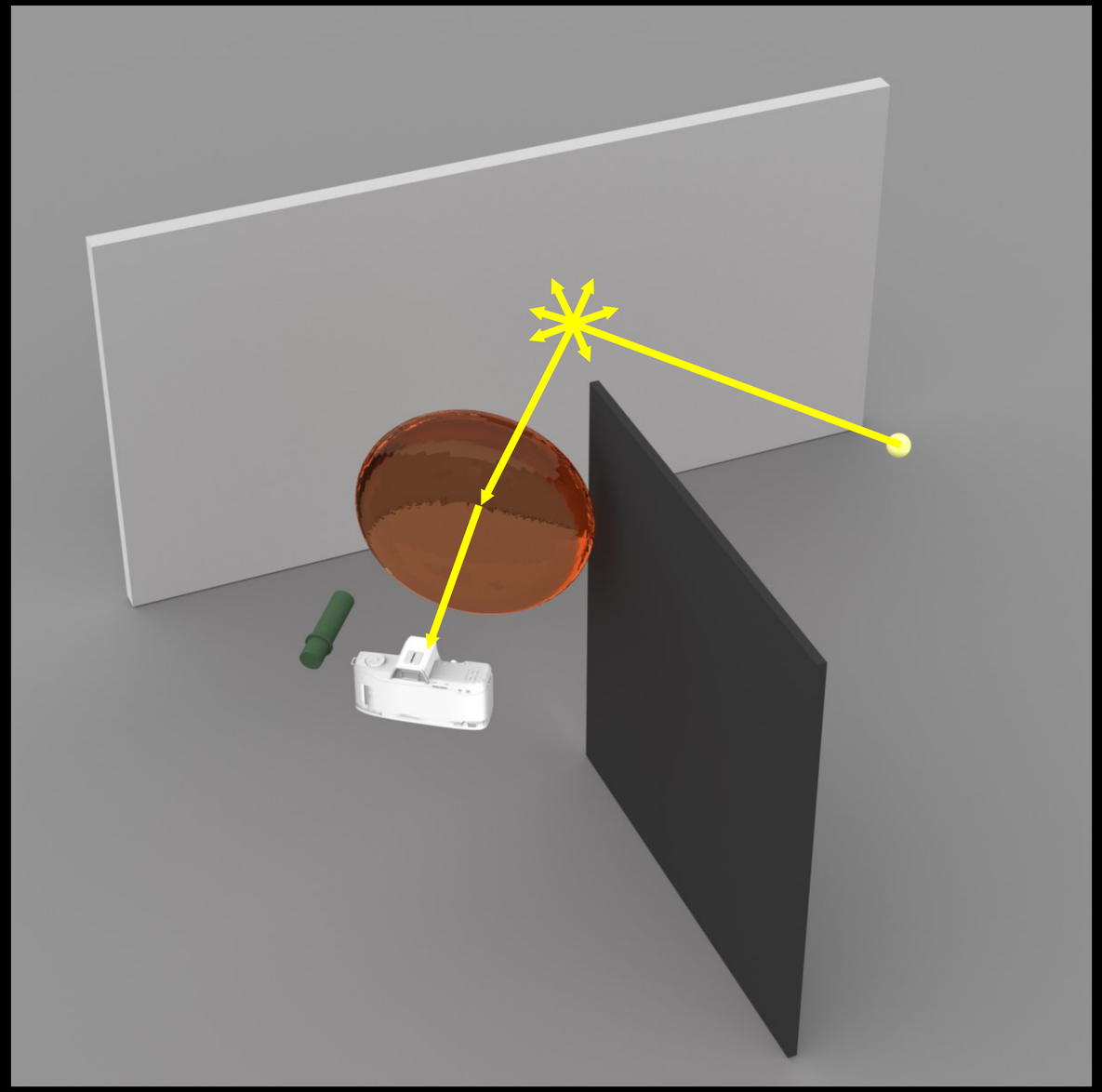
How do we focus on a voxel?

challenge:

non-specular photons

solution:

use a large lens?
expensive !!



How do we focus on a voxel?

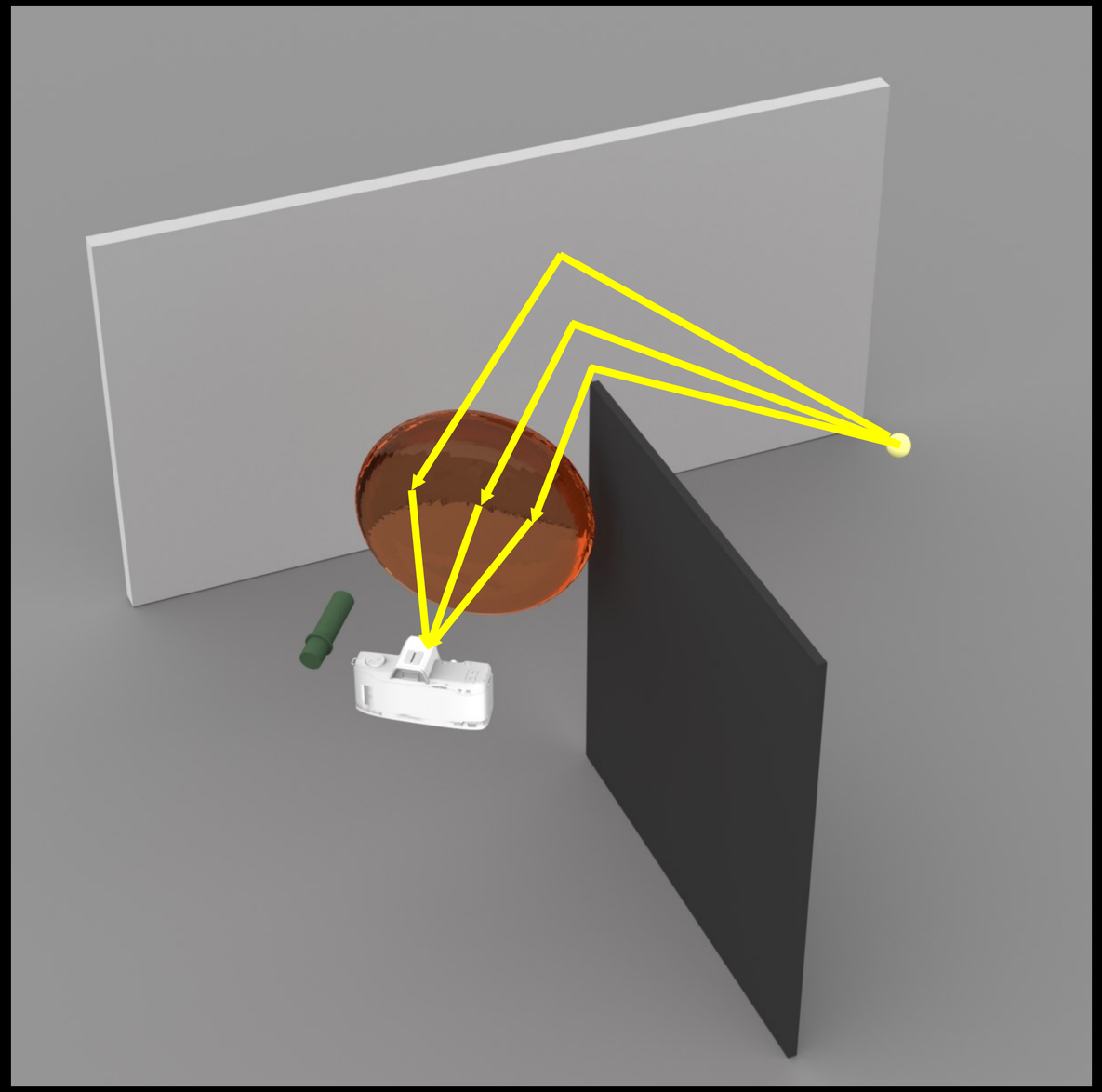
challenge:

non-specular photons

solution:

use a large lens?
expensive !!

what does a lens do?
delays rays such that they reach detector at same time instant



Temporal focusing: imitate the lens

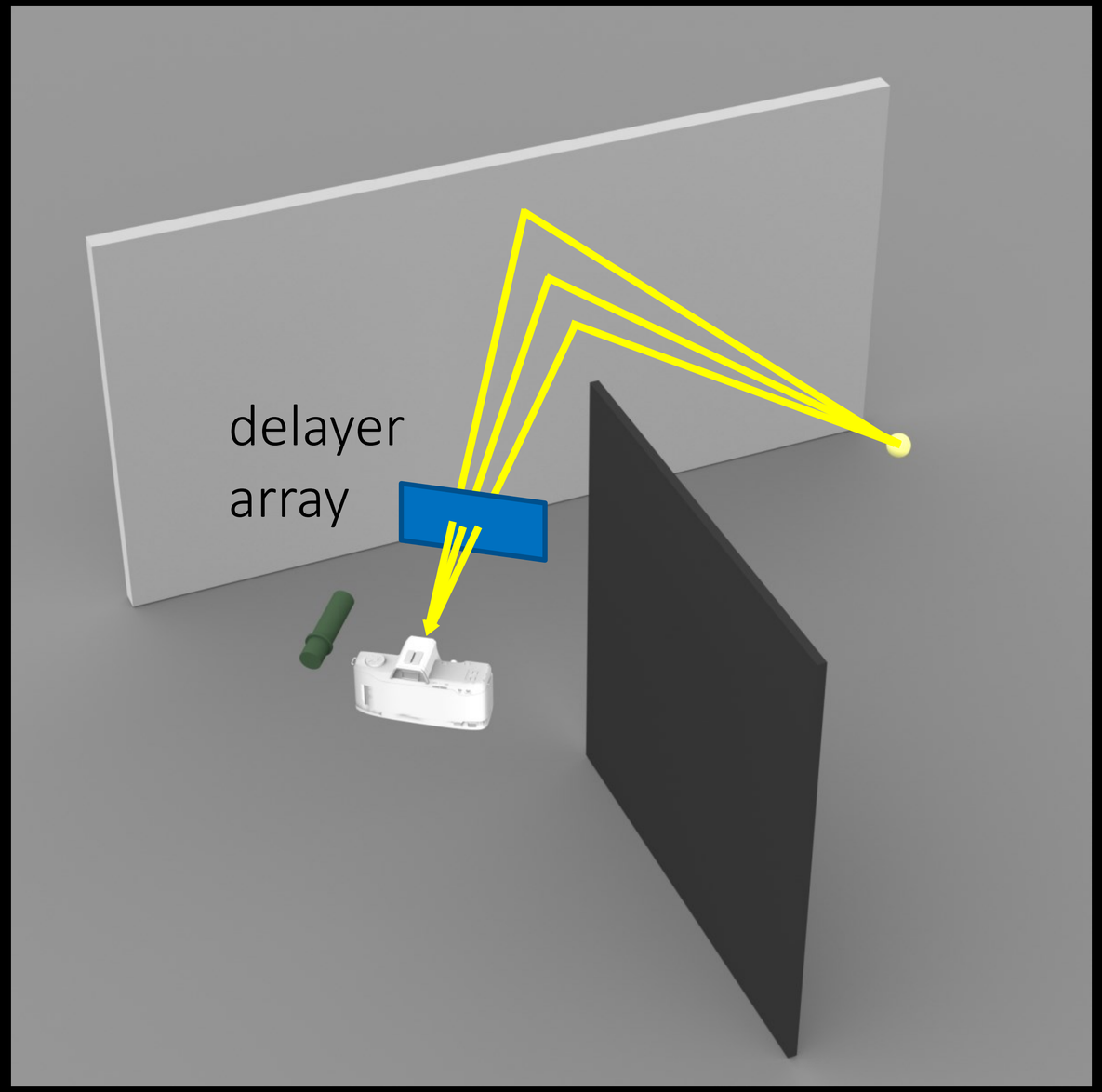
challenge:

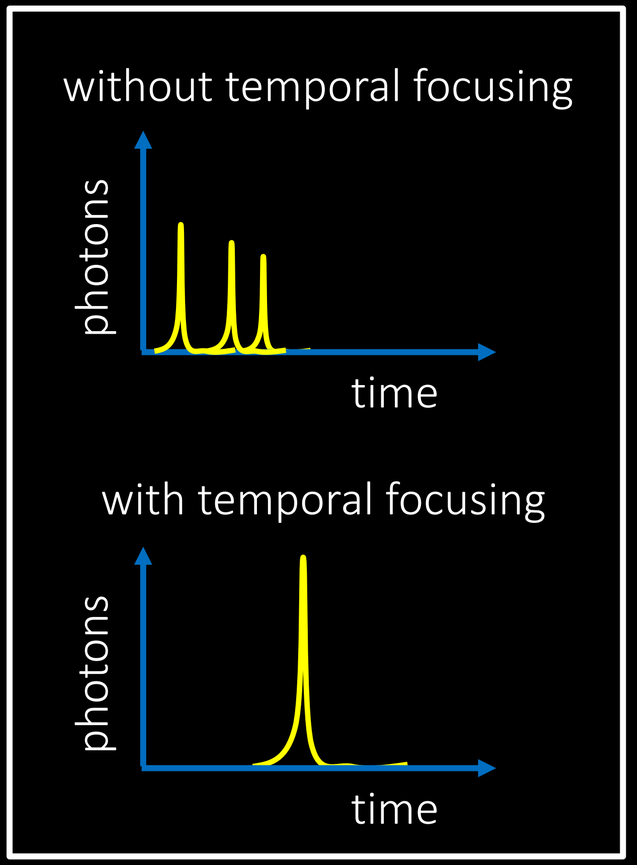
non-specular photons

solution:

use a large lens?
expensive !!

temporal focusing:
imitate large lens





Temporal focusing: imitate the lens

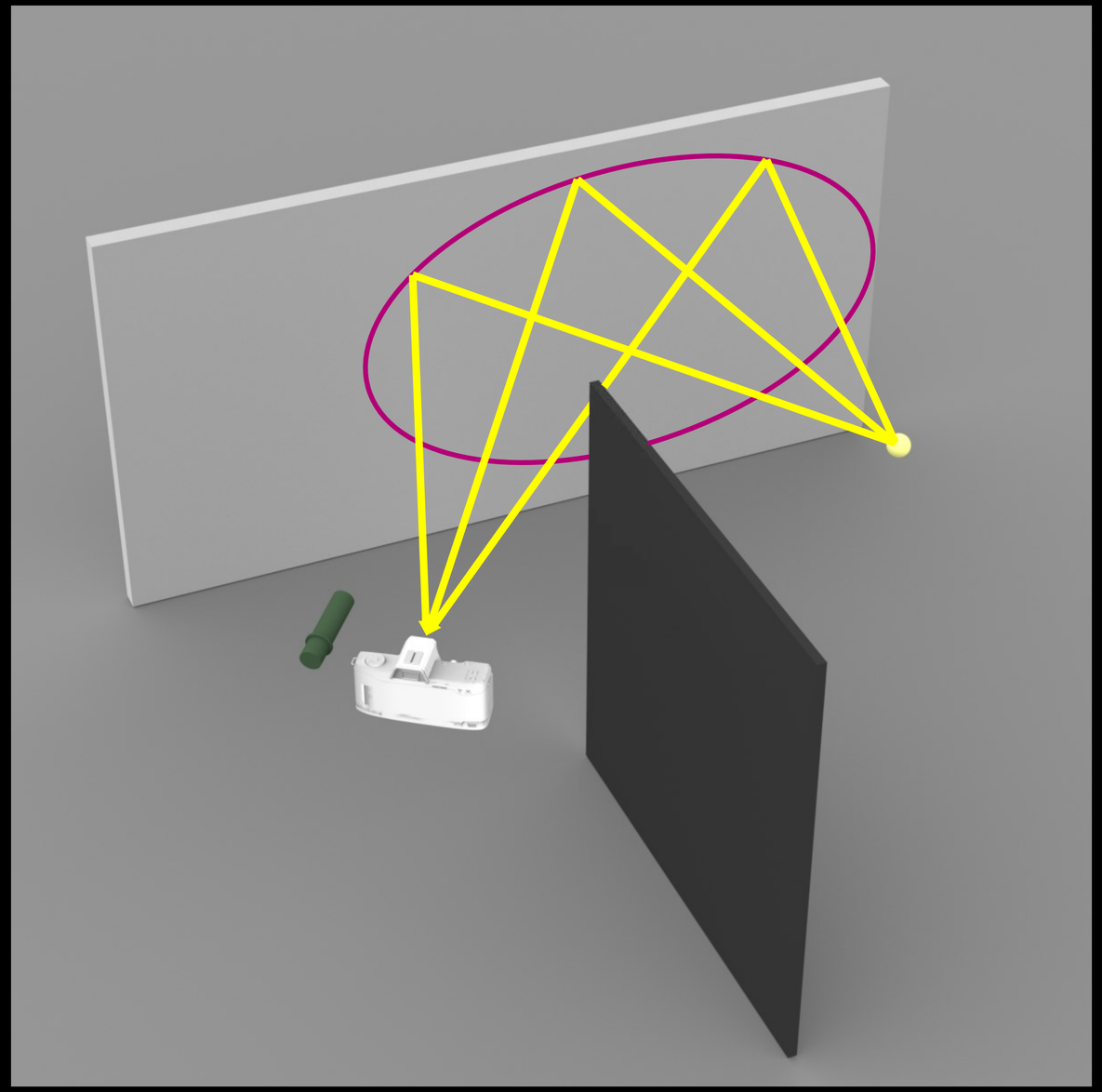
challenge:

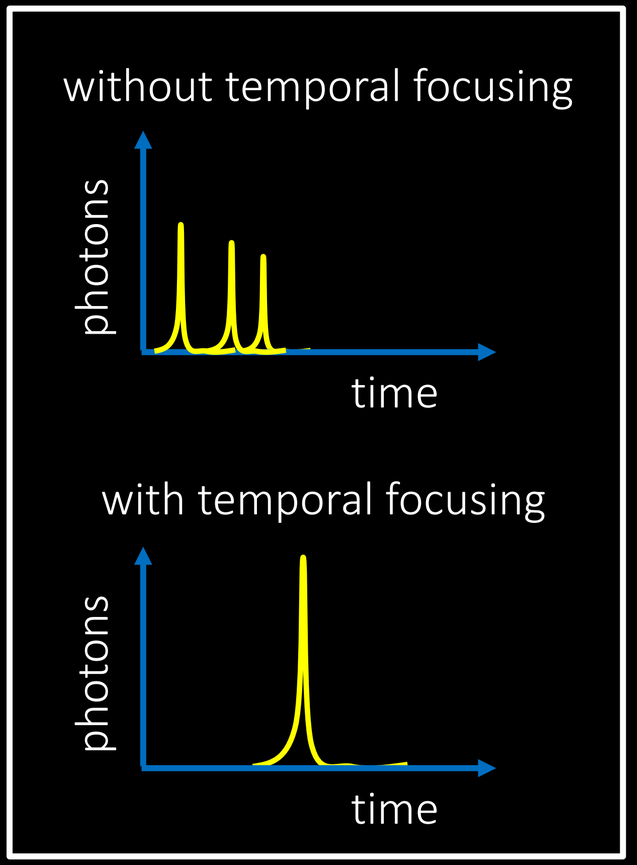
non-specular photons

solution:

use a large lens?
expensive !!

temporal focusing:
imitate large lens





Temporal focusing: Illumination should also be an ellipse

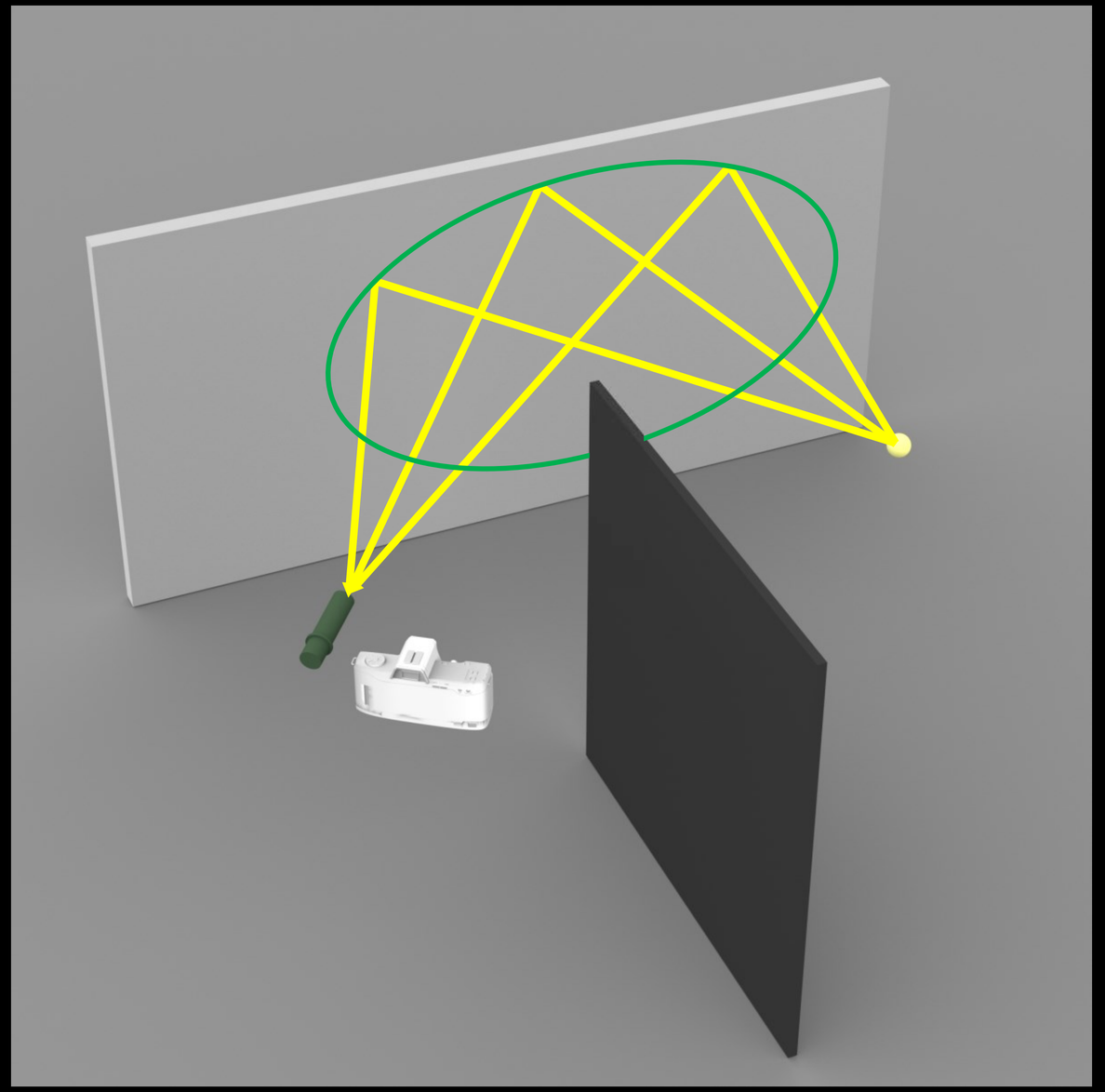
challenge:

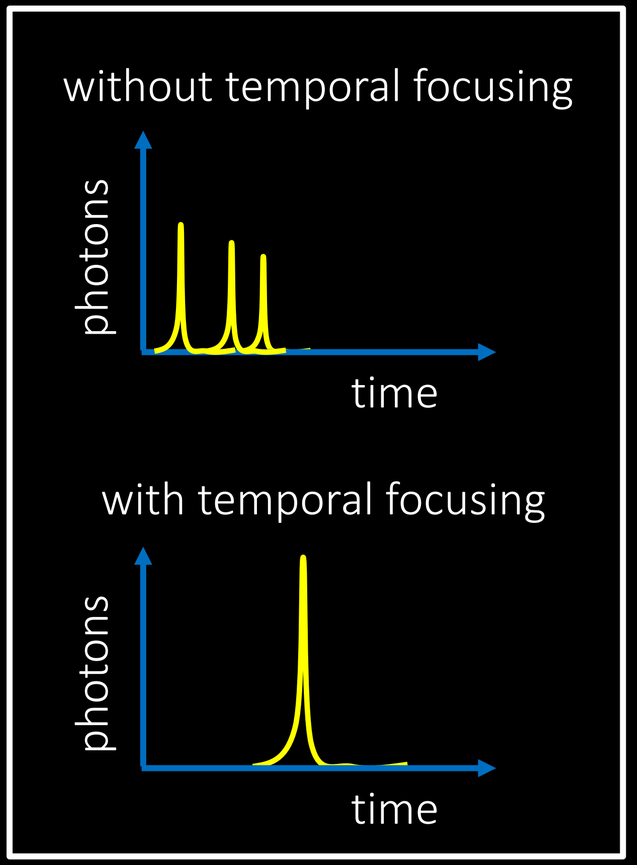
non-specular photons

solution:

use a large lens?
expensive !!

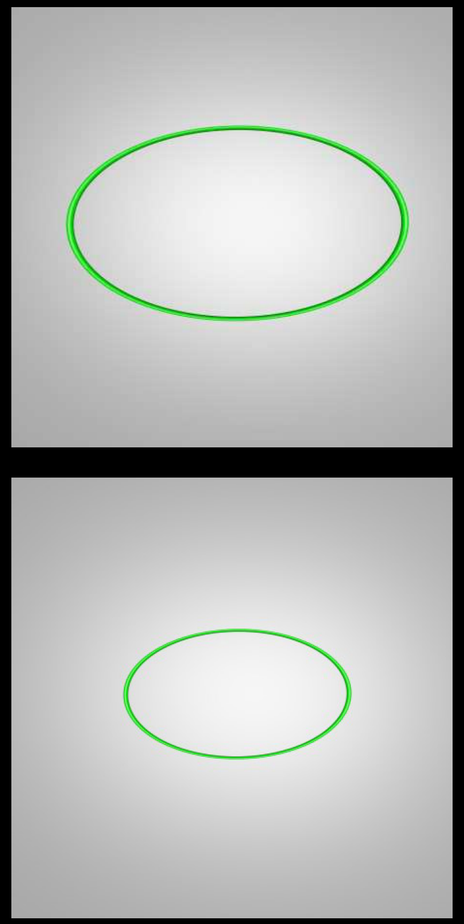
temporal focusing:
imitate large lens



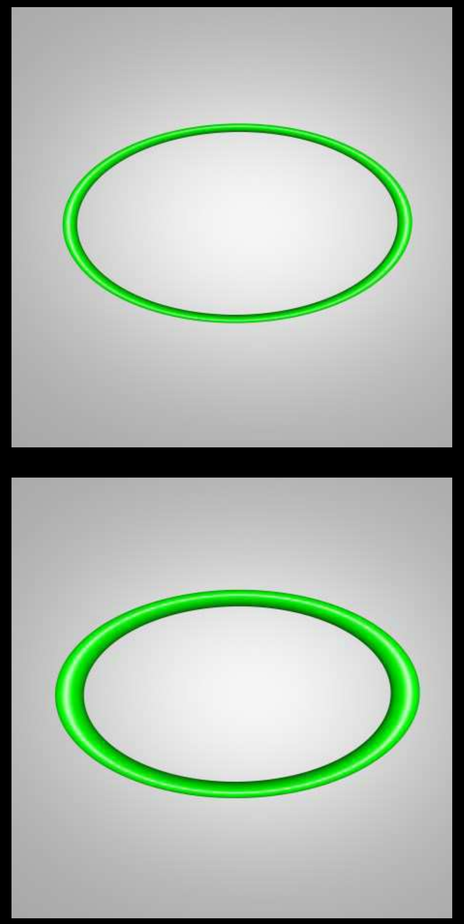


Design choices for temporal focusing

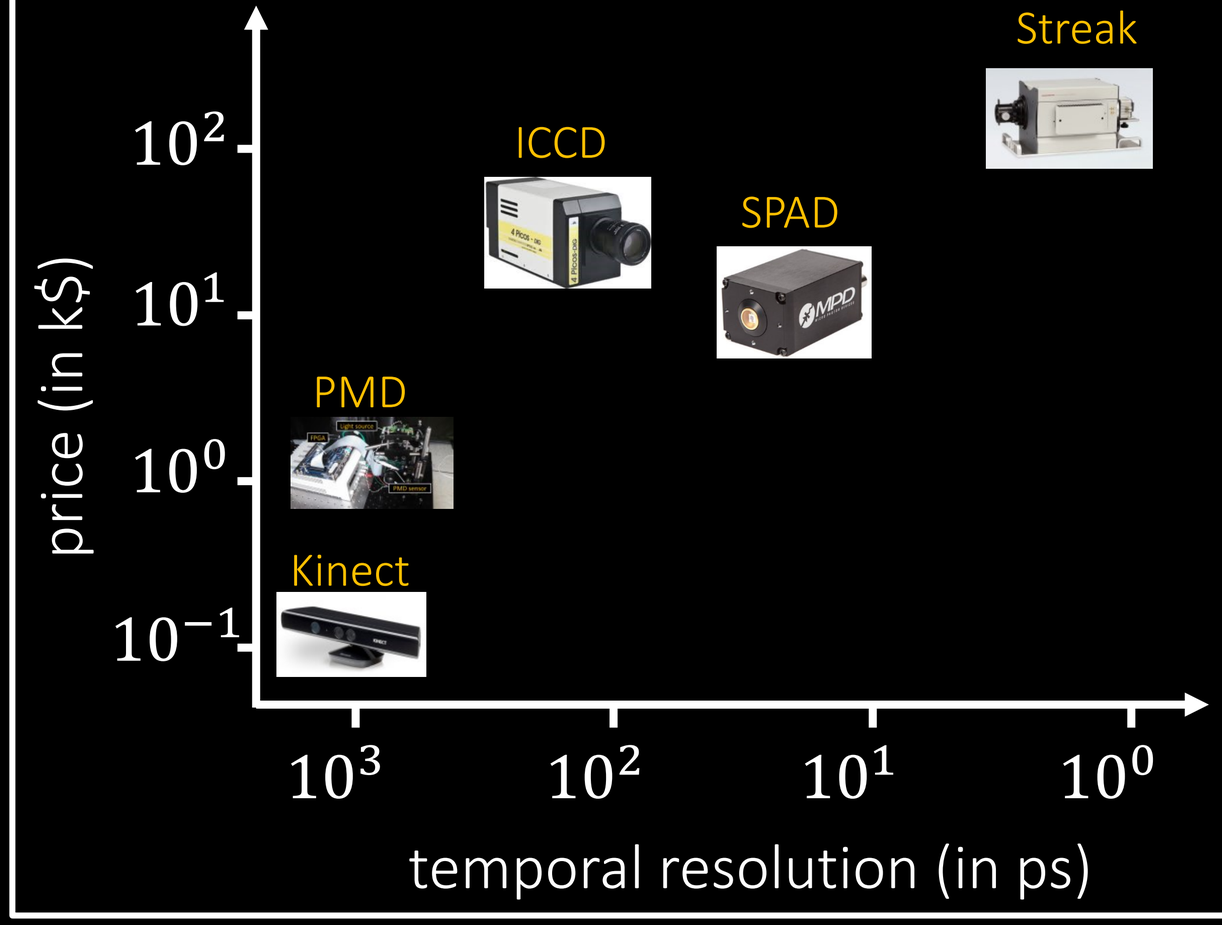
ellipse size



ellipse thickness

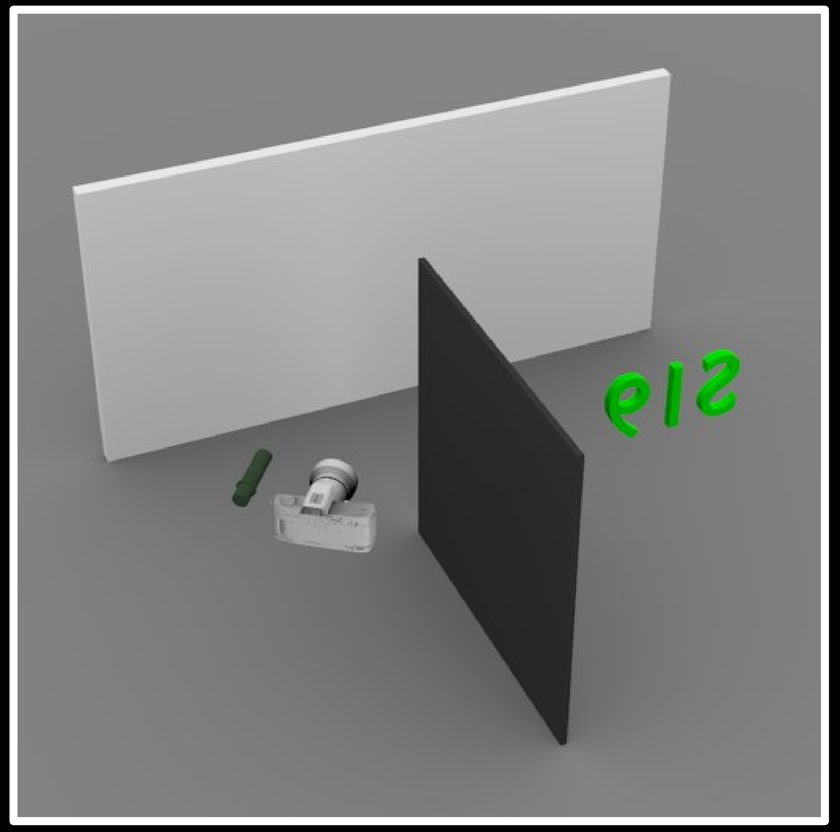


time-of-flight cameras

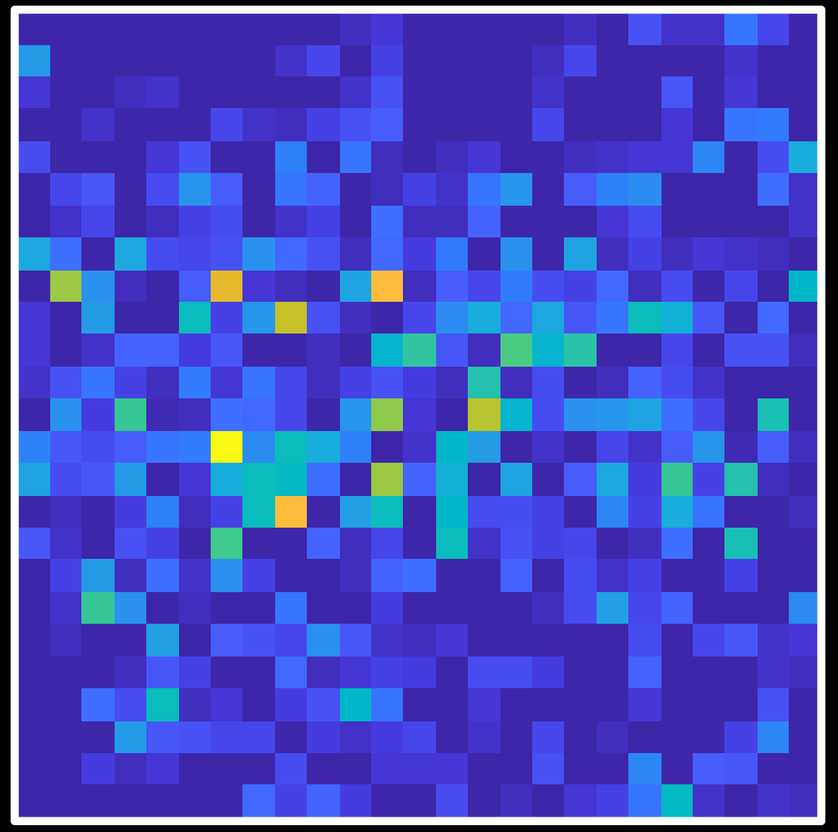


Pediredla et al. ICCP 2019.

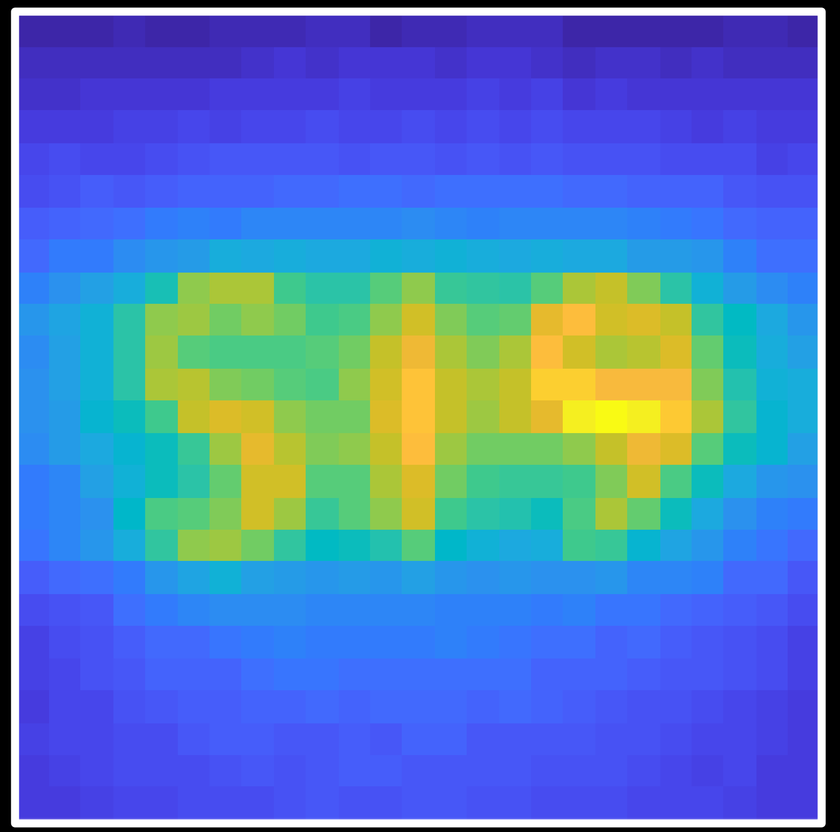
Rendering non-line-of-sight imaging by temporal focusing



scene



standard BDPT



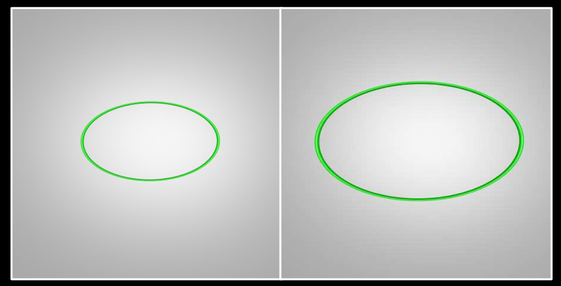
BDPT w/ ellipsoidal connections

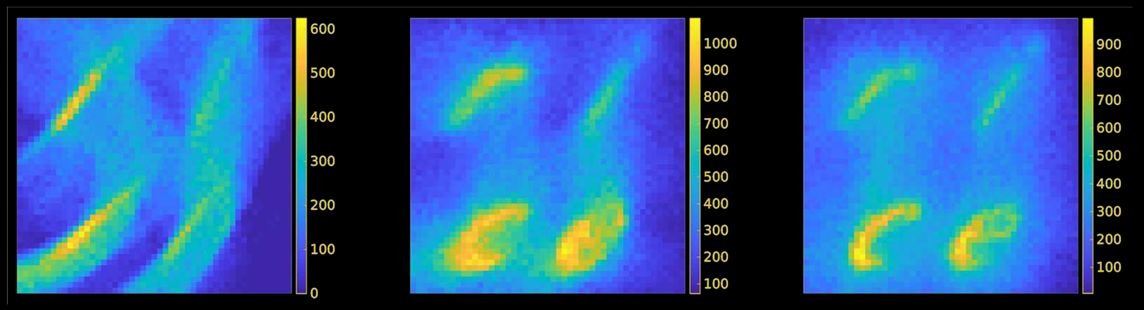
Gate width: 4 ps (0.4% scene)
Rendering time: 3 hr

∗ simulation results

Design choices

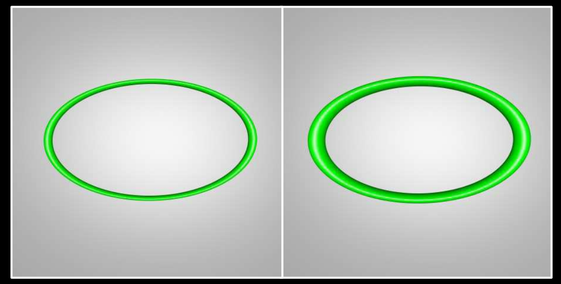
ellipse size

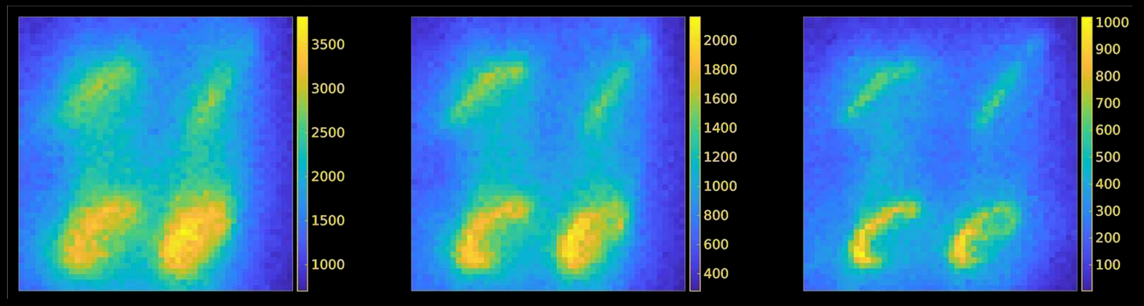




large ellipses result in better resolution

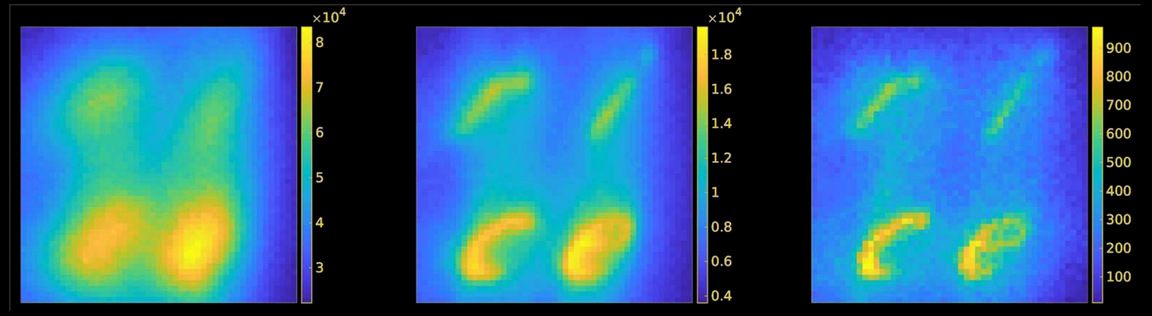
ellipse thickness





thinner ellipses result in better resolution, but looses light

temporal resolution



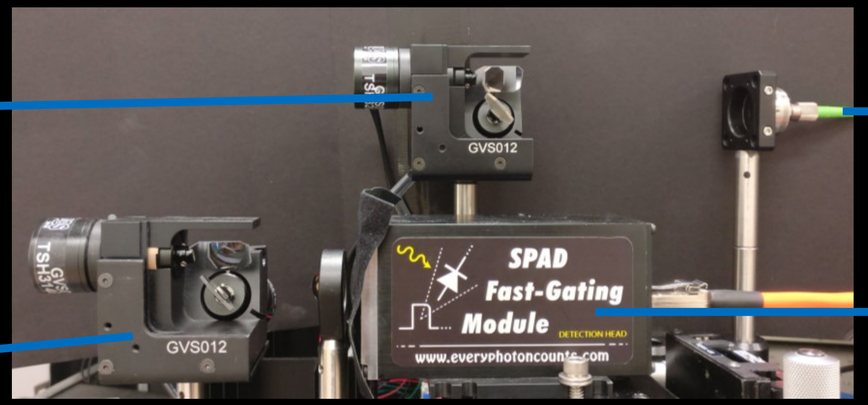
high resolution time-gate results in better resolution, but looses light

Hardware prototype

optical component

illumination galvo

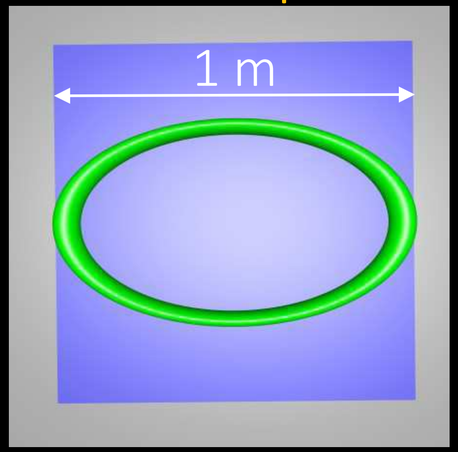
imaging galvo



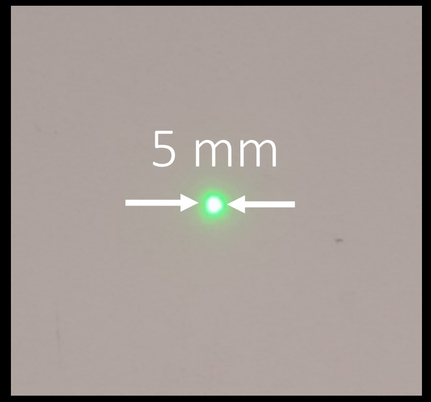
picosecond laser

SPAD

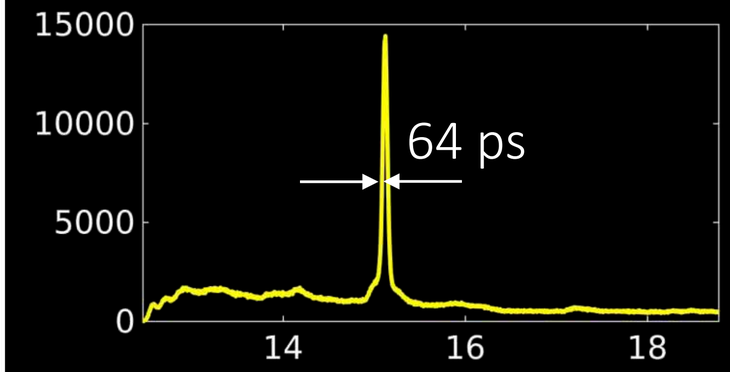
Size of ellipses



Thickness of ellipse (laser spot size)



Measured Jitter



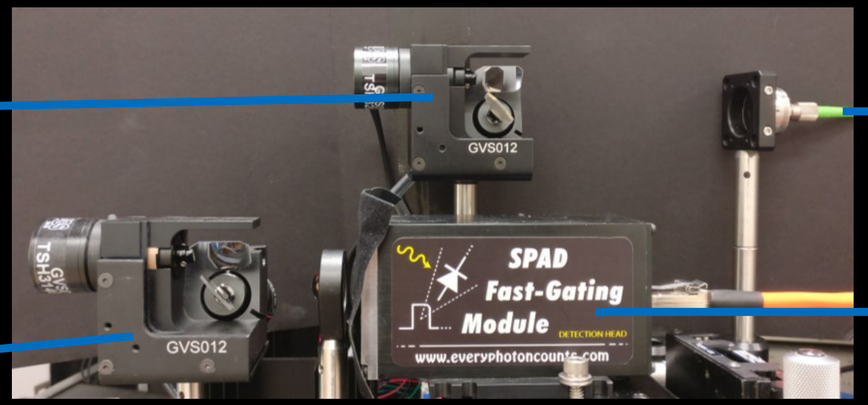
Pediredla et al. ICCP 2019.

Hardware prototype

optical component

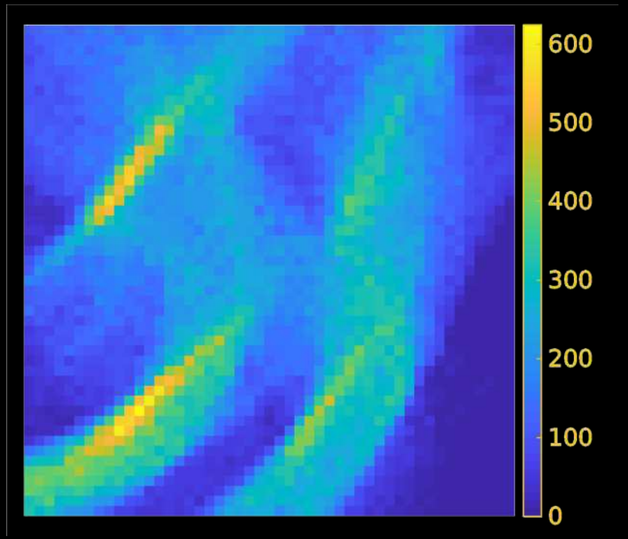
illumination galvo

imaging galvo



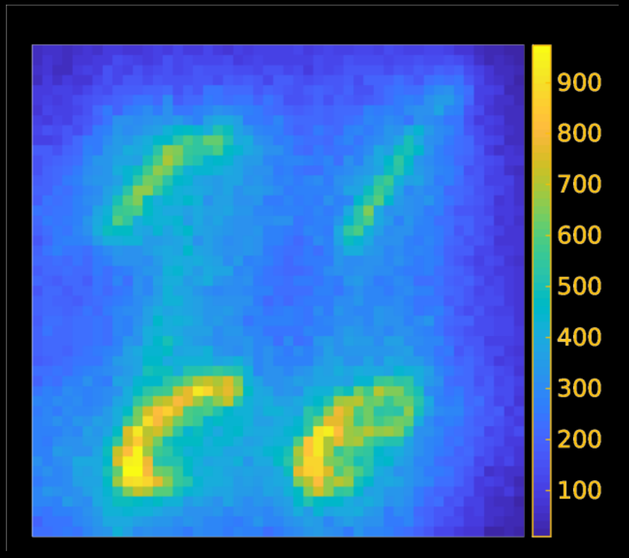
picosecond laser

SPAD



unoptimized

renderer driven optimization



optimized

Pediredla et al. ICCP 2019.

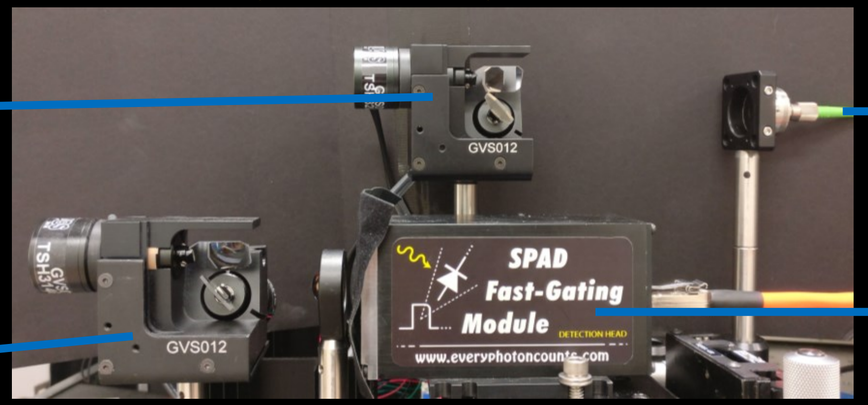
Hardware prototype

optical component

MEMS
Axicon
SLM
OPA

illumination galvo

imaging galvo

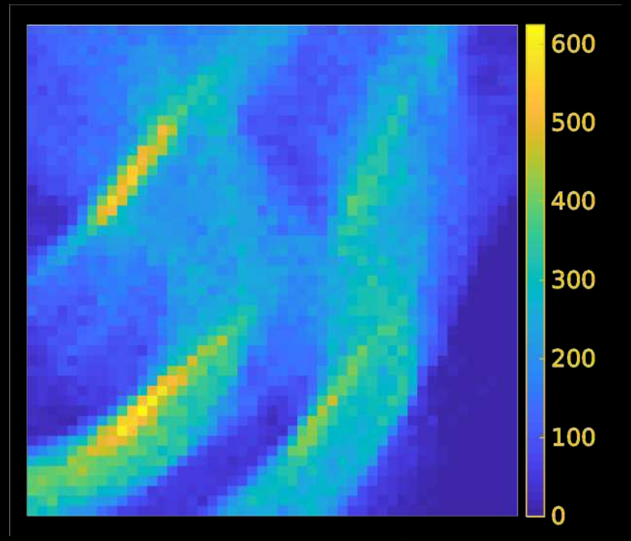


picosecond laser

LED
SLED
supercontinuum
femtosecond laser

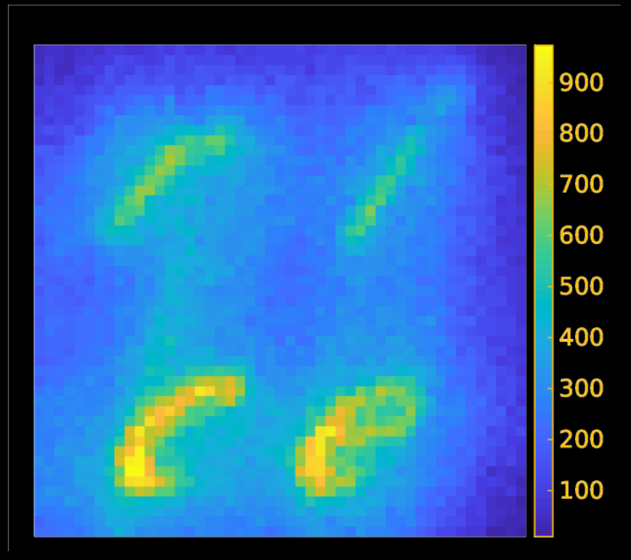
SPAD

streak
ICCD
kinect
PMD



unoptimized

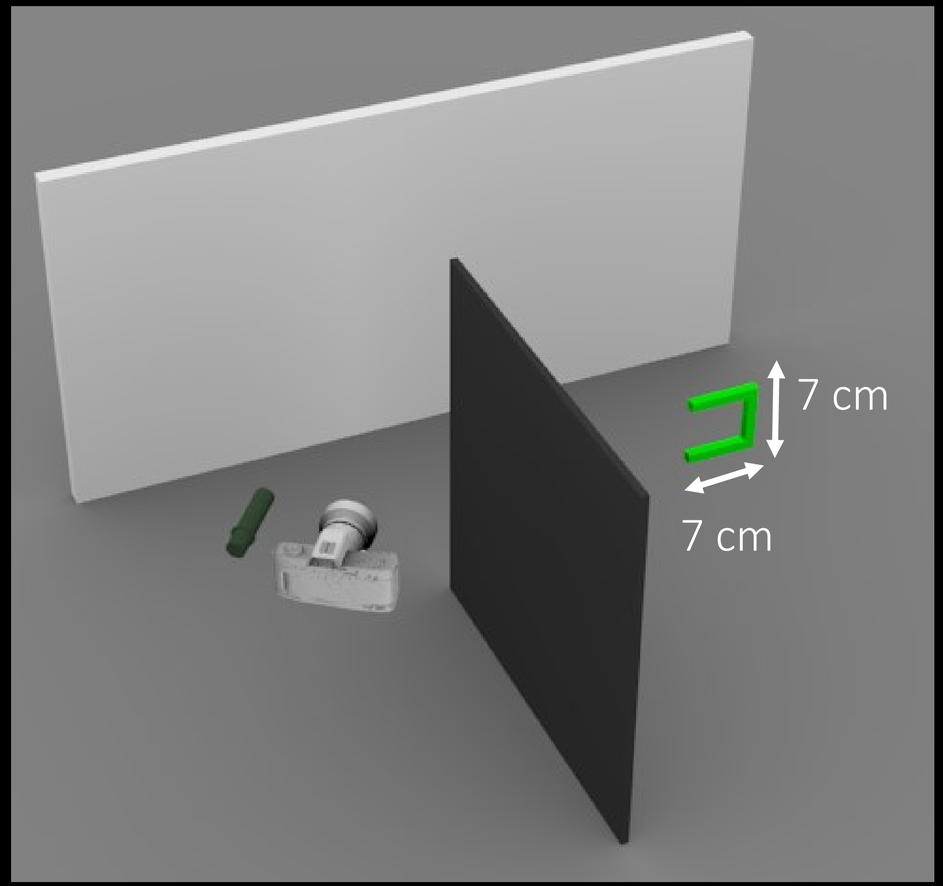
renderer driven optimization



optimized

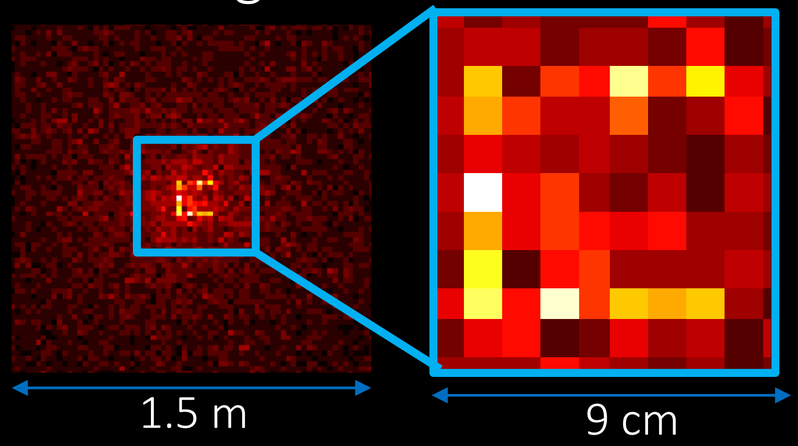
Pediredla et al. ICCP 2019.

Results: scanning limited ROI



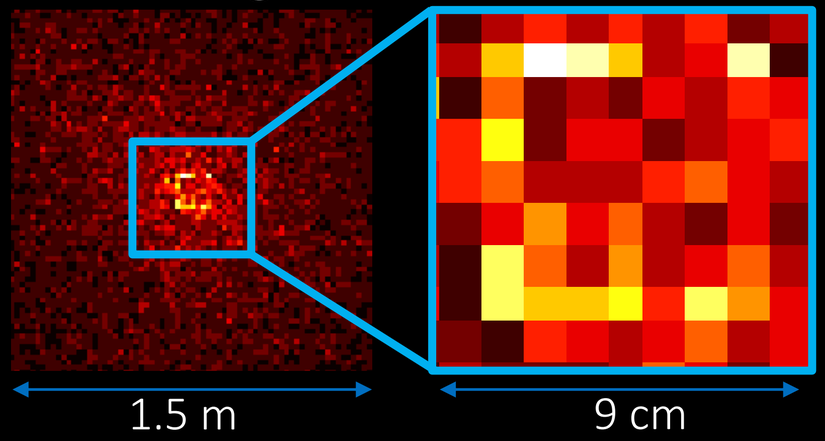
previous techniques

scanning entire hidden scene

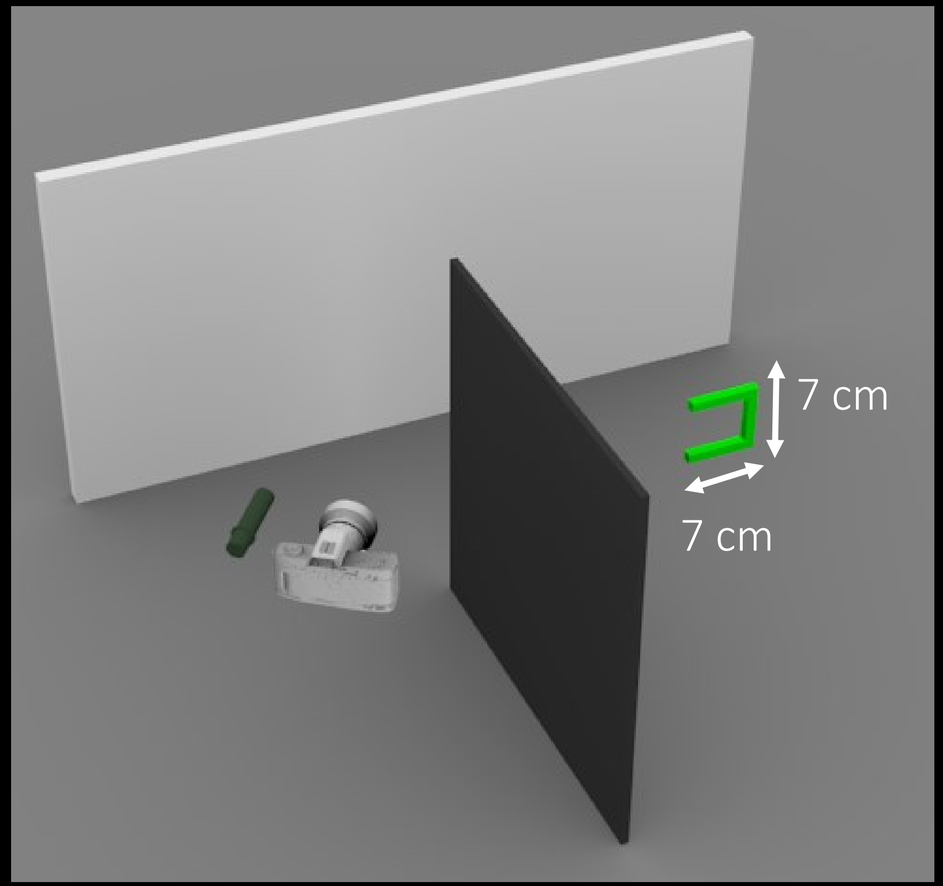


temporal focusing

scanning entire hidden scene



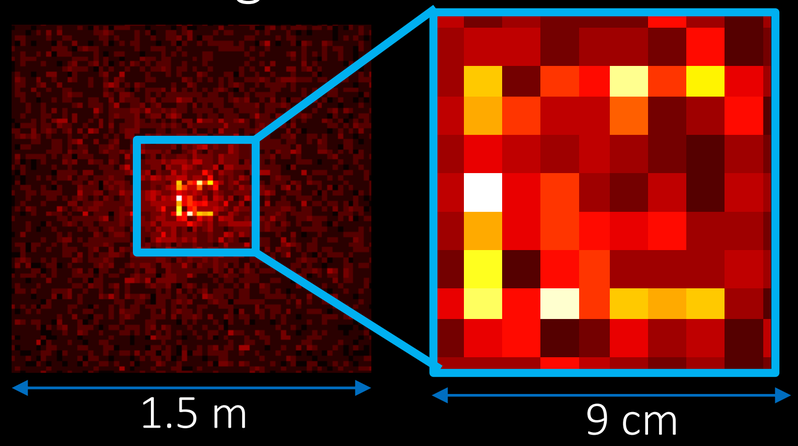
Results: scanning limited ROI



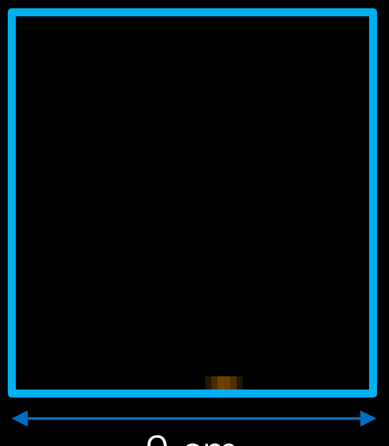
SNR of temporal focusing is
> 10× higher for small ROI

previous techniques

scanning entire hidden scene

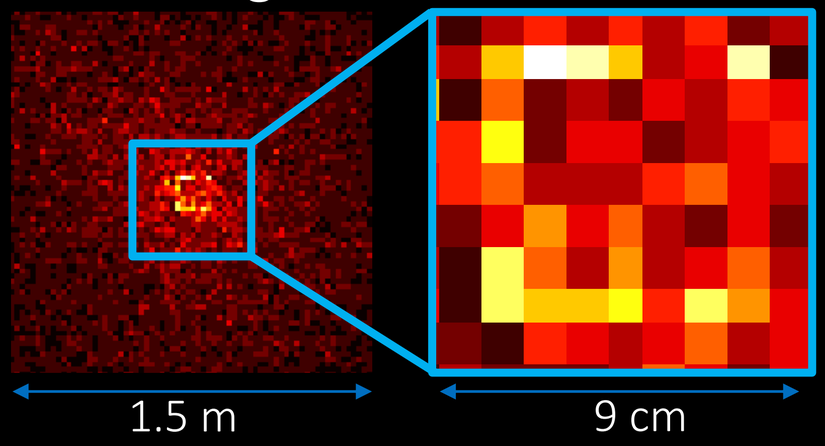


cannot scan ROI

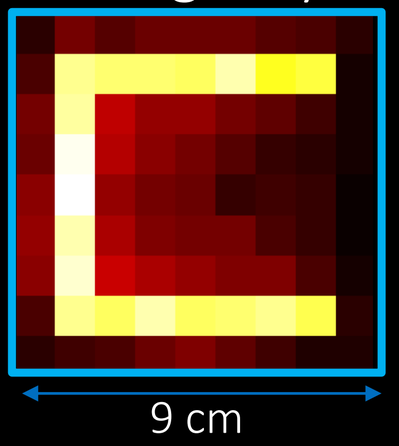


temporal focusing

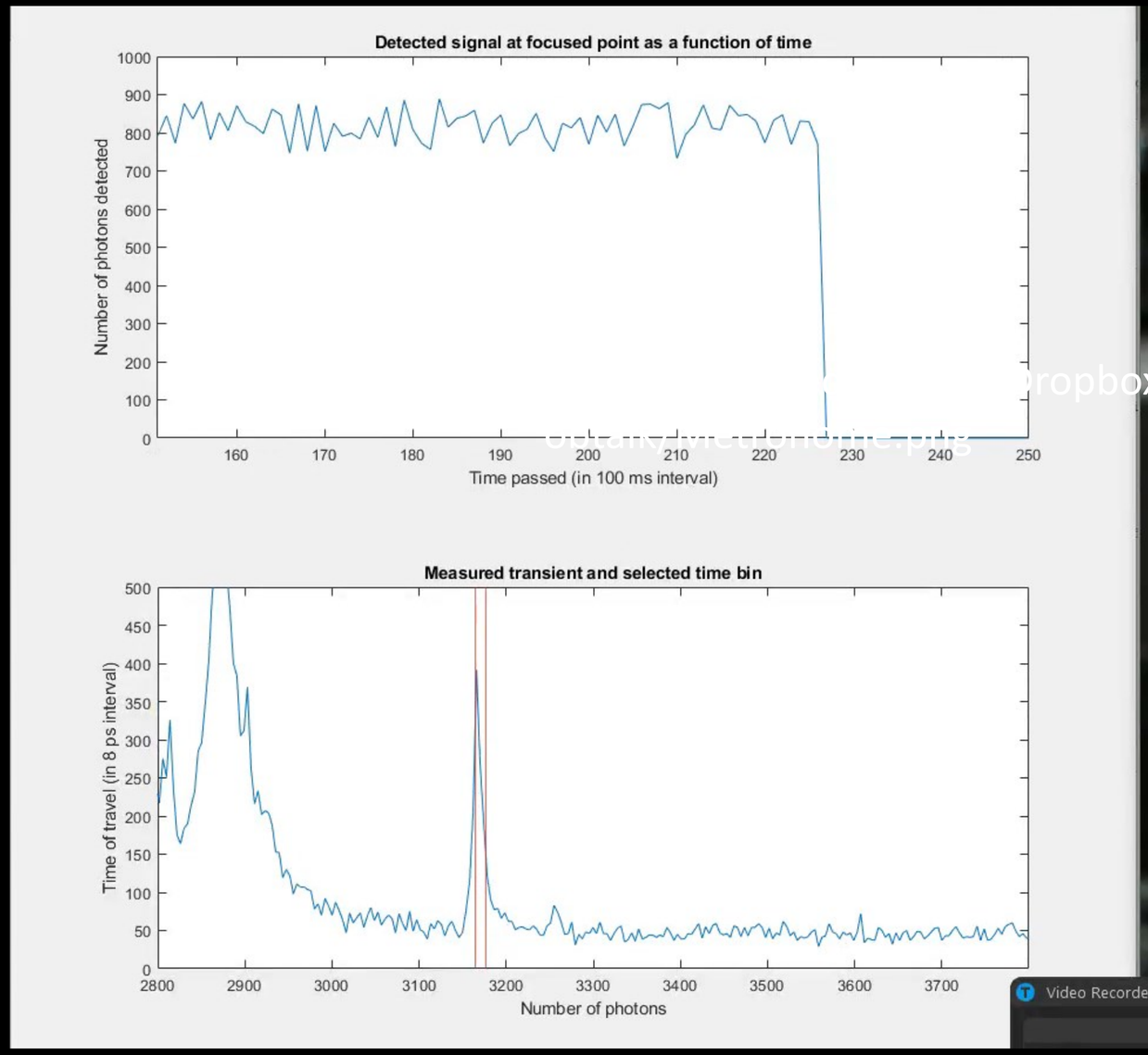
scanning entire hidden scene

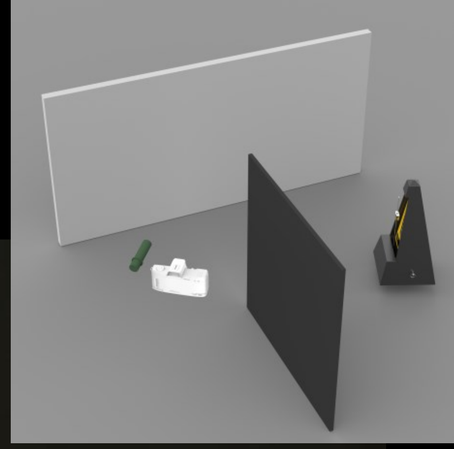


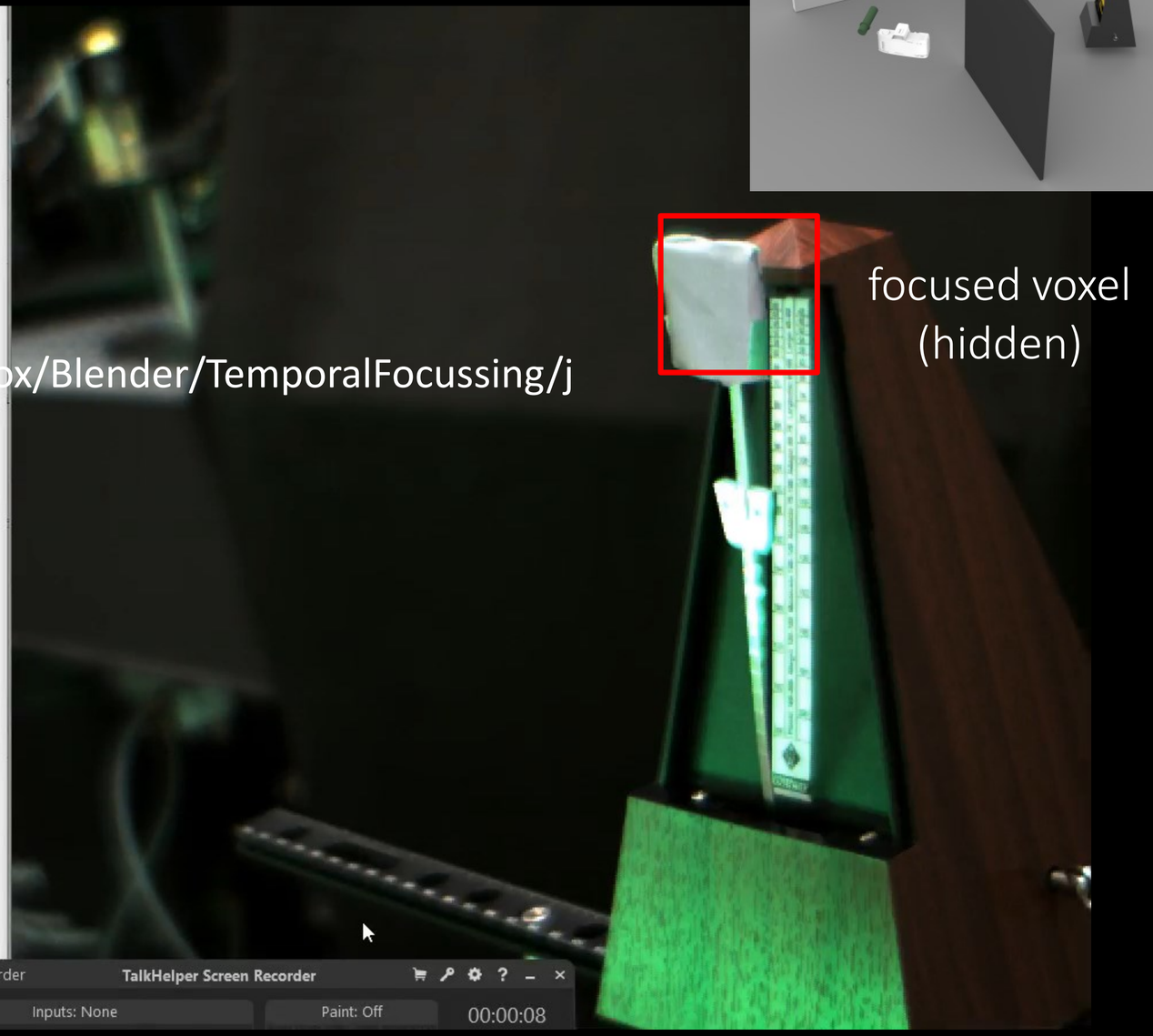
scanning only ROI



Results: real-time occupancy detection







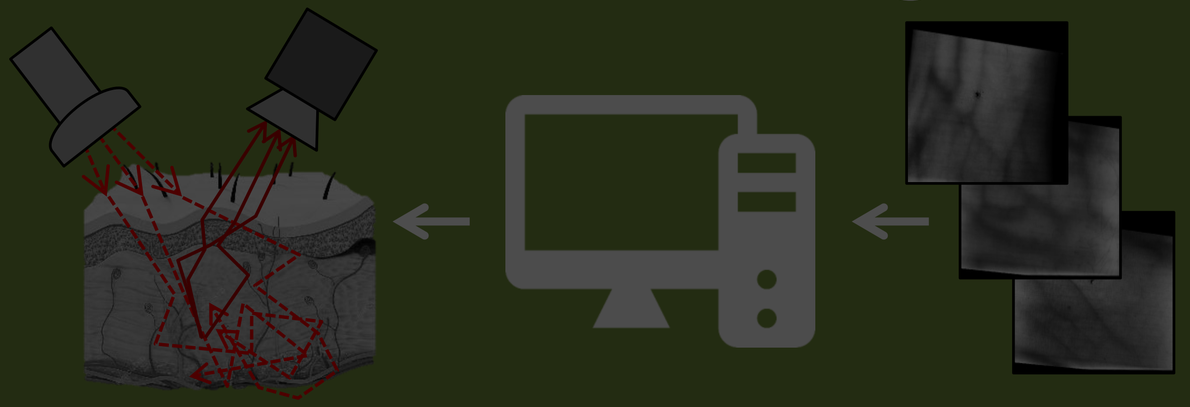
focused voxel (hidden)

Physics-based rendering and its applications to computational imaging

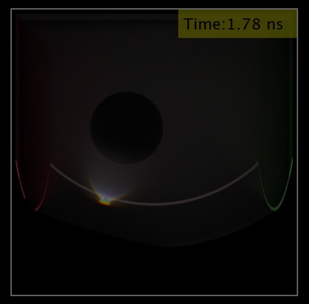
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

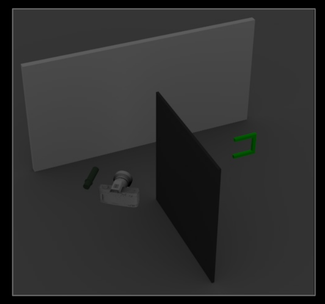
inverse rendering



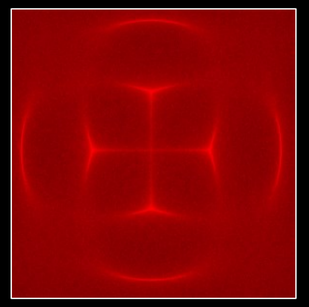
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



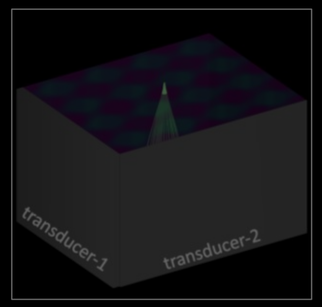
time-of-flight imaging



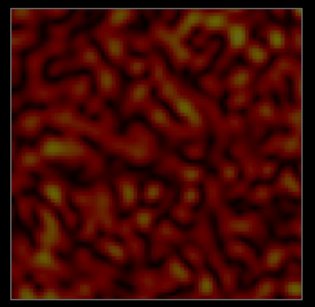
non-line-of-sight imaging



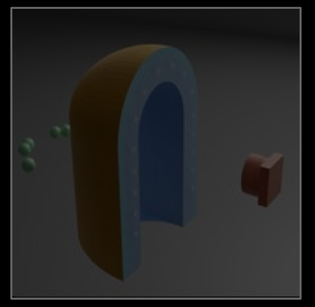
acousto-optic lensing



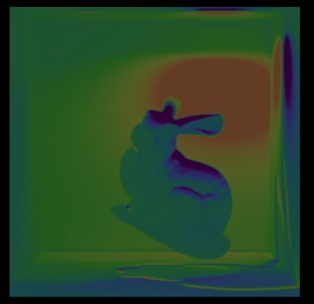
ultrafast light scanning



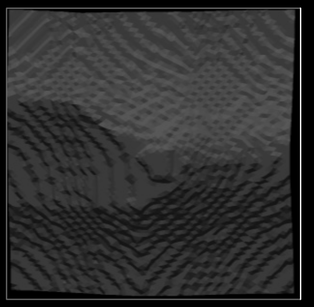
speckle imaging



tactile sensor design

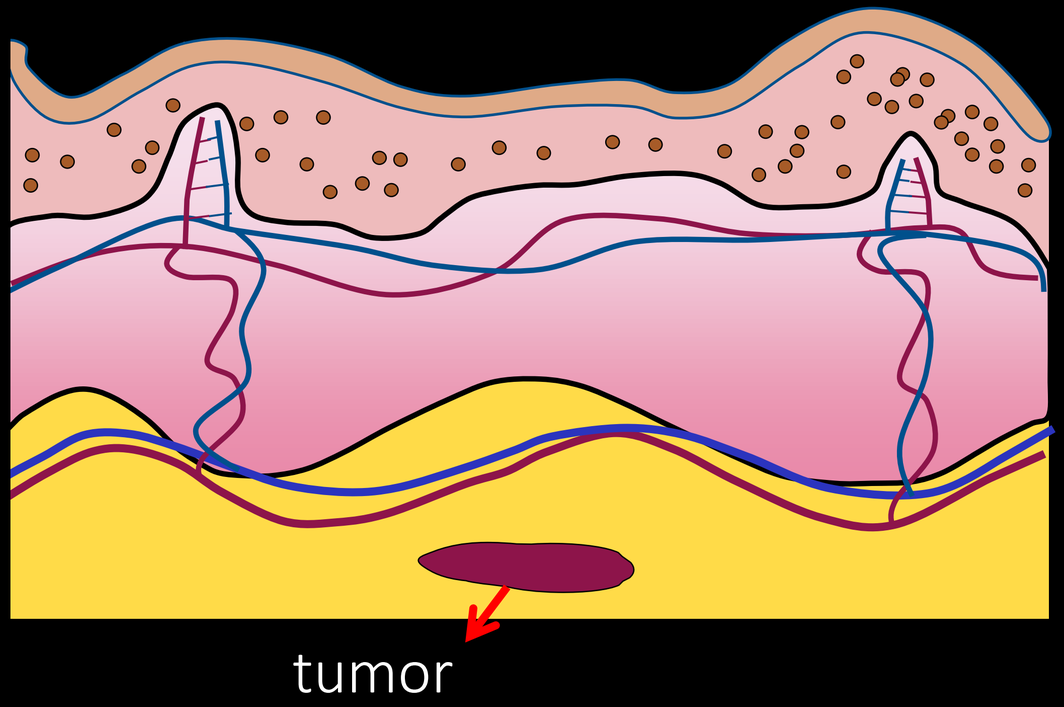


differentiable renderer

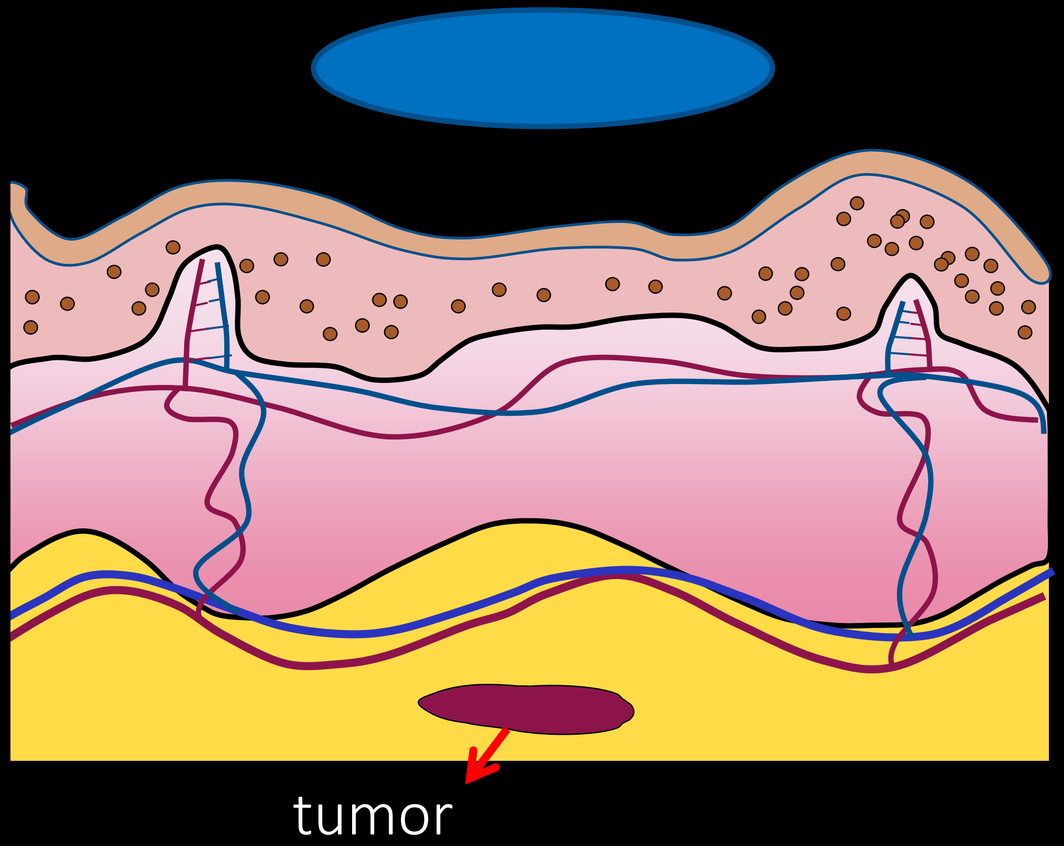


inverse problems

Focusing light inside tissue



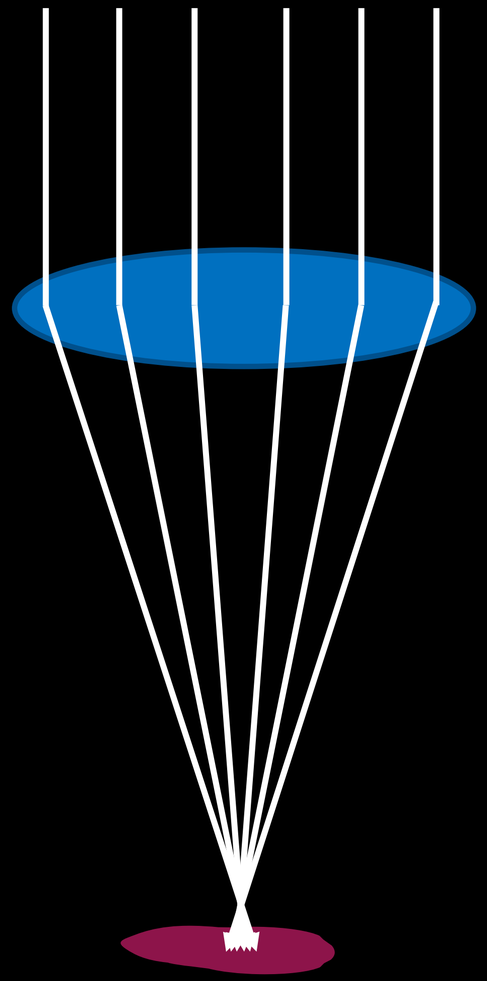
Focusing light inside tissue



Focusing light inside tissue

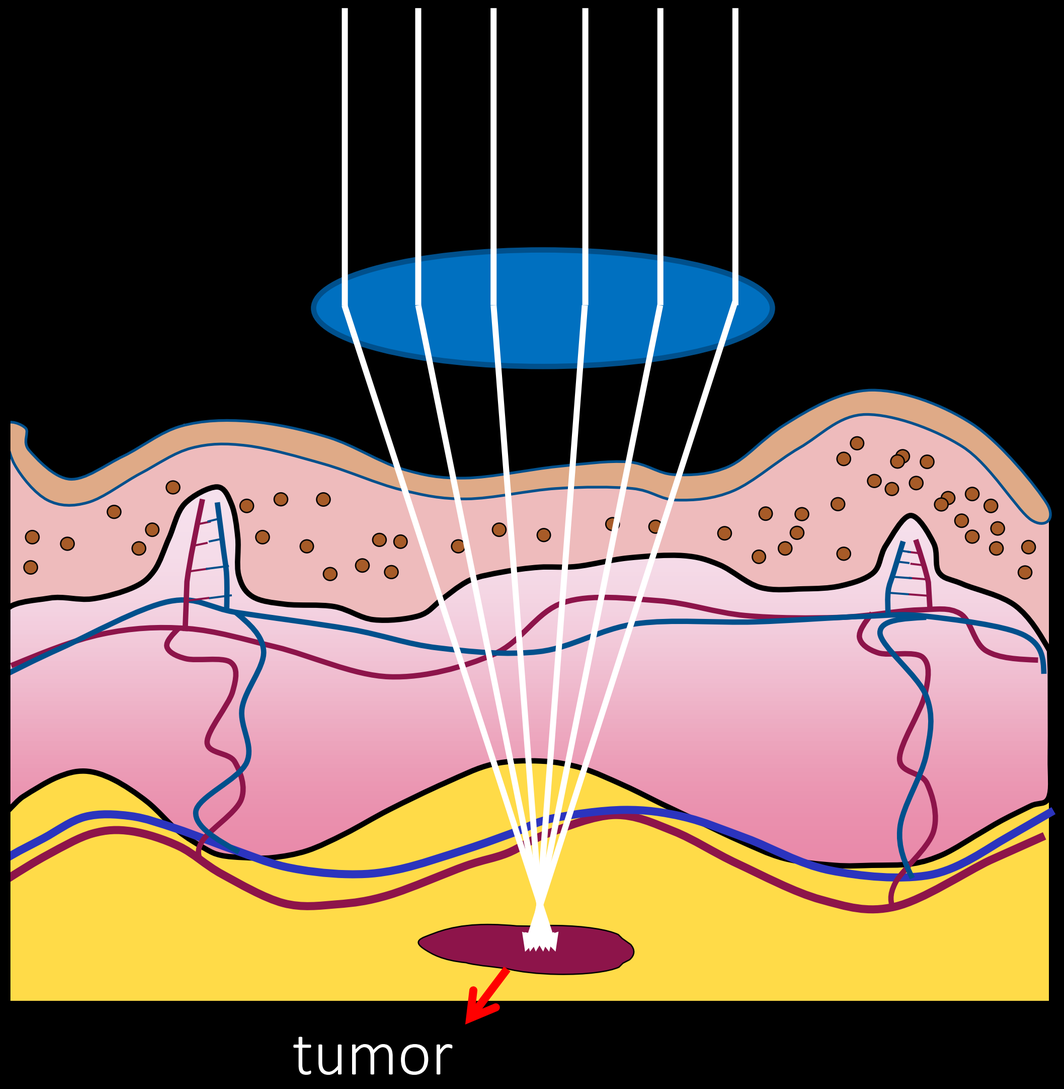
tumor

Focusing light inside tissue

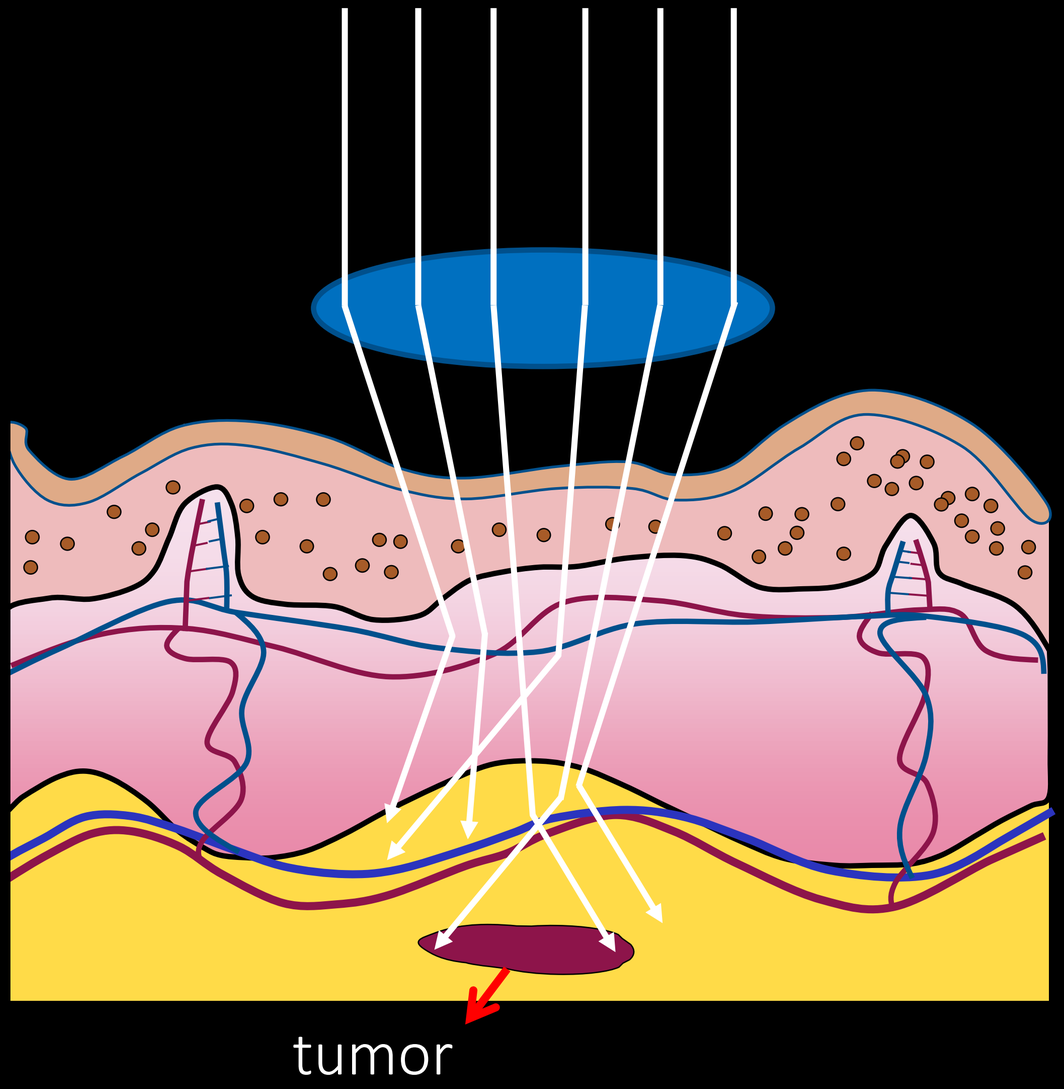


tumor

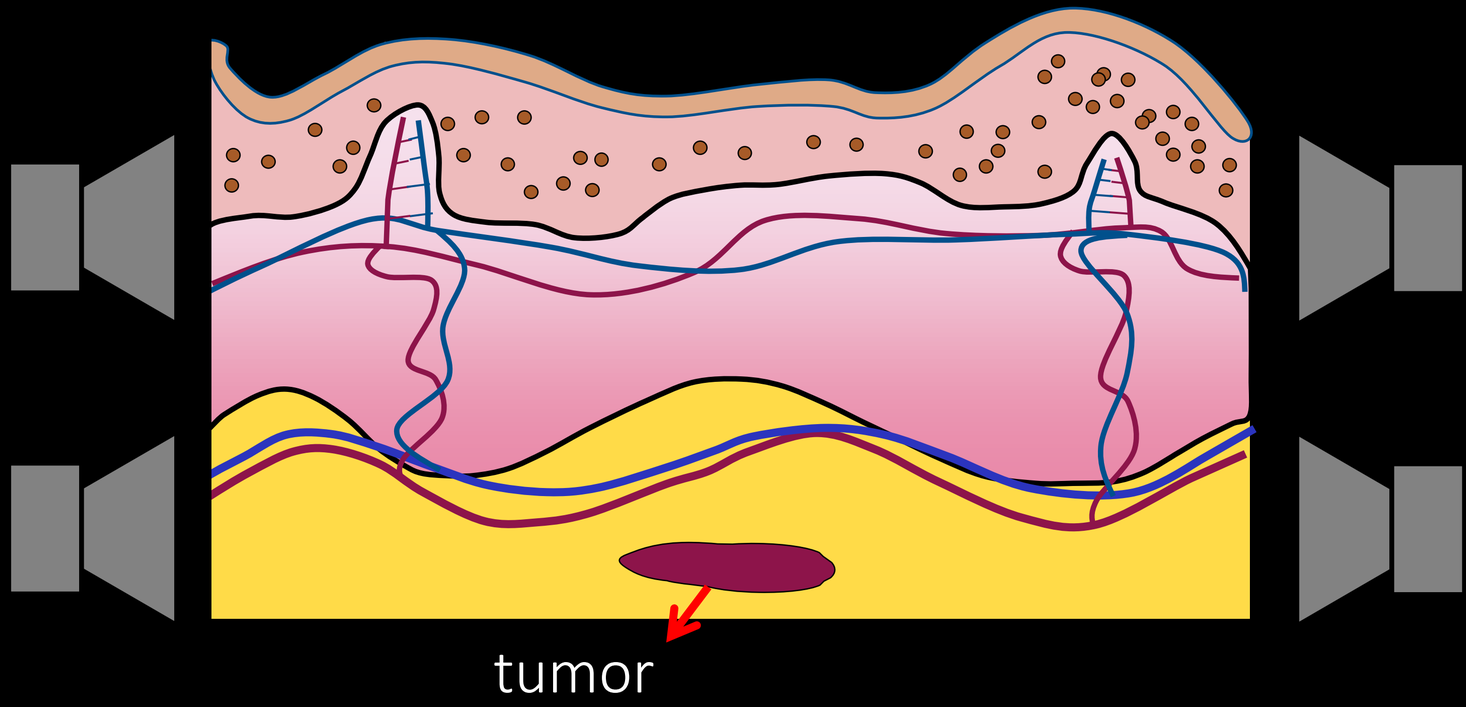
Focusing light inside tissue



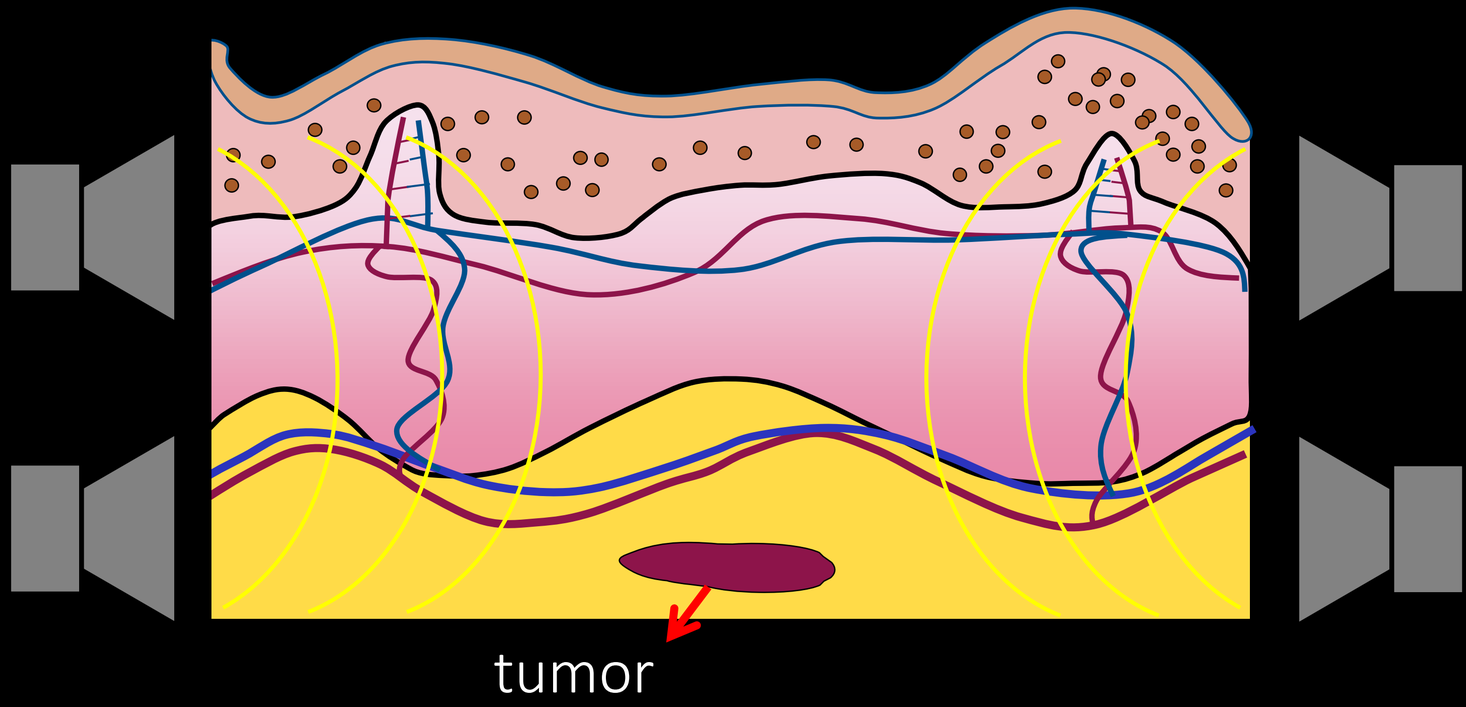
Focusing light inside tissue



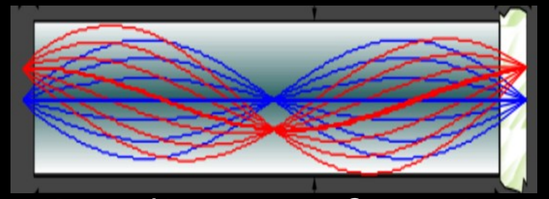
Focusing light inside tissue



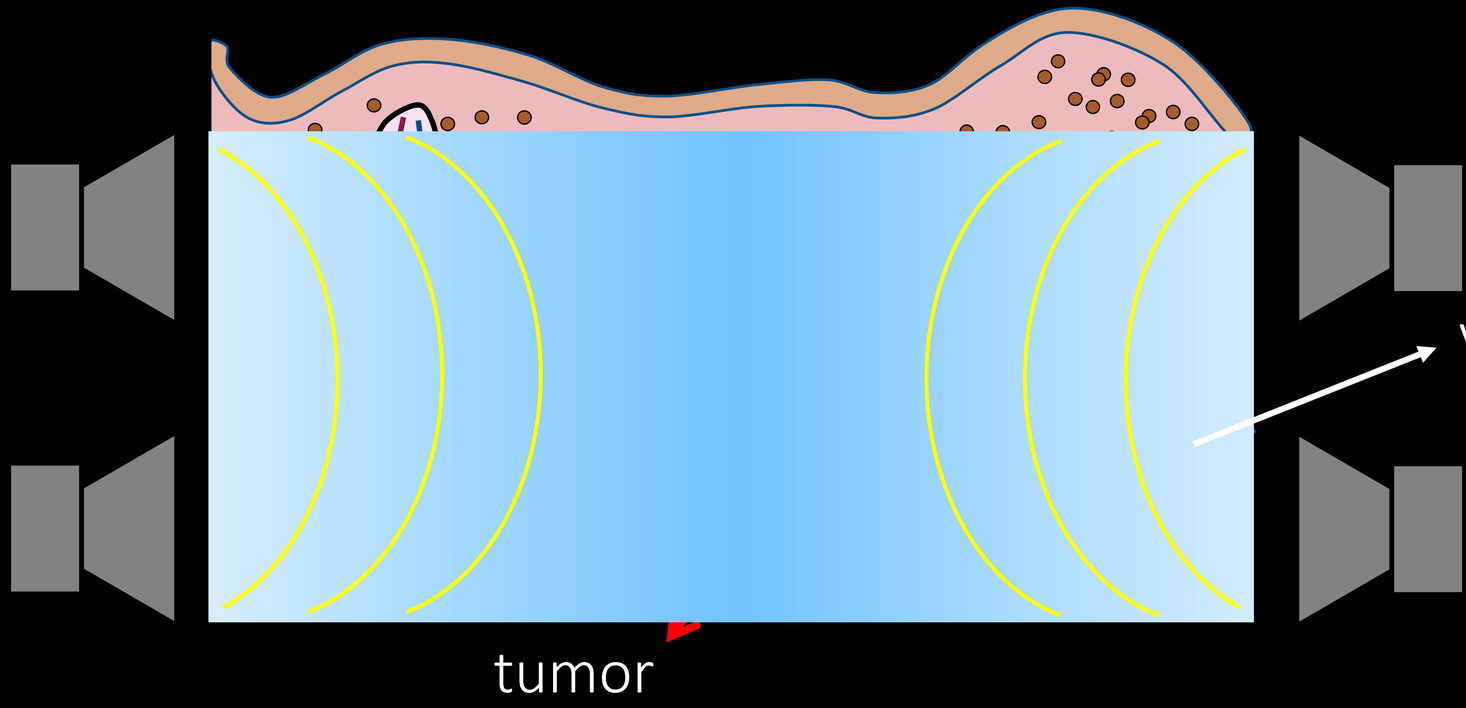
Focusing light inside tissue



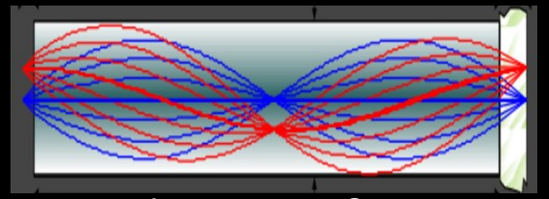
Focusing light inside tissue



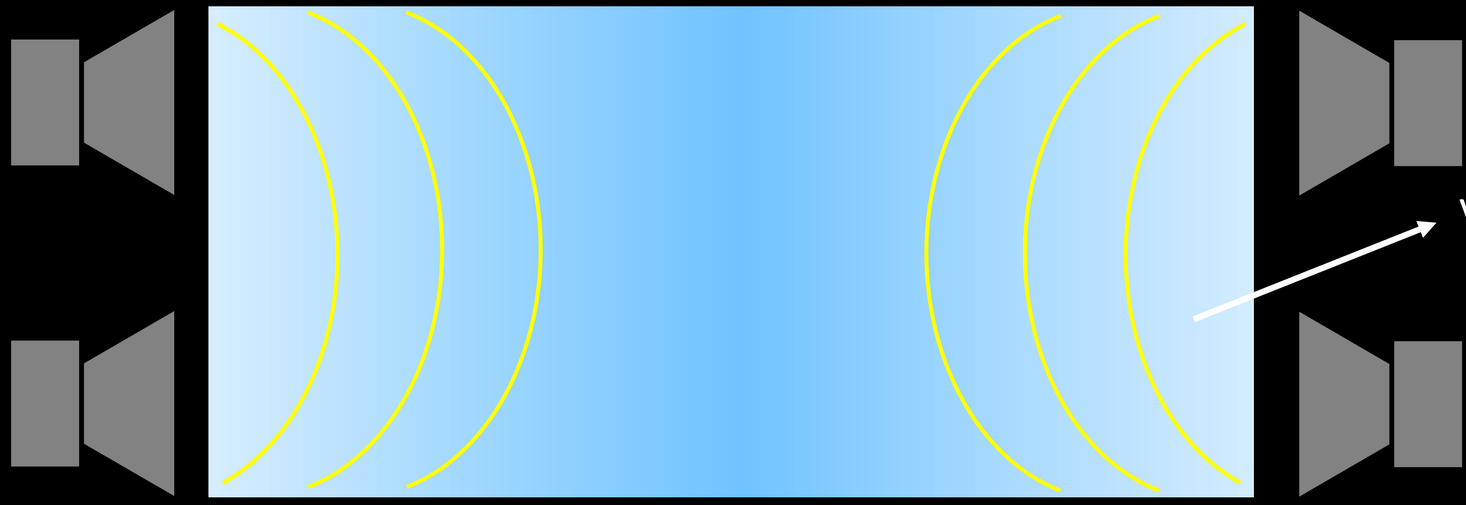
Gradient Refractive Index (GRIN) waveguide



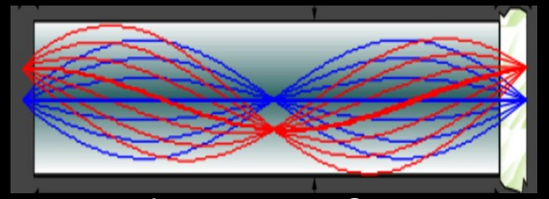
Focusing light inside tissue



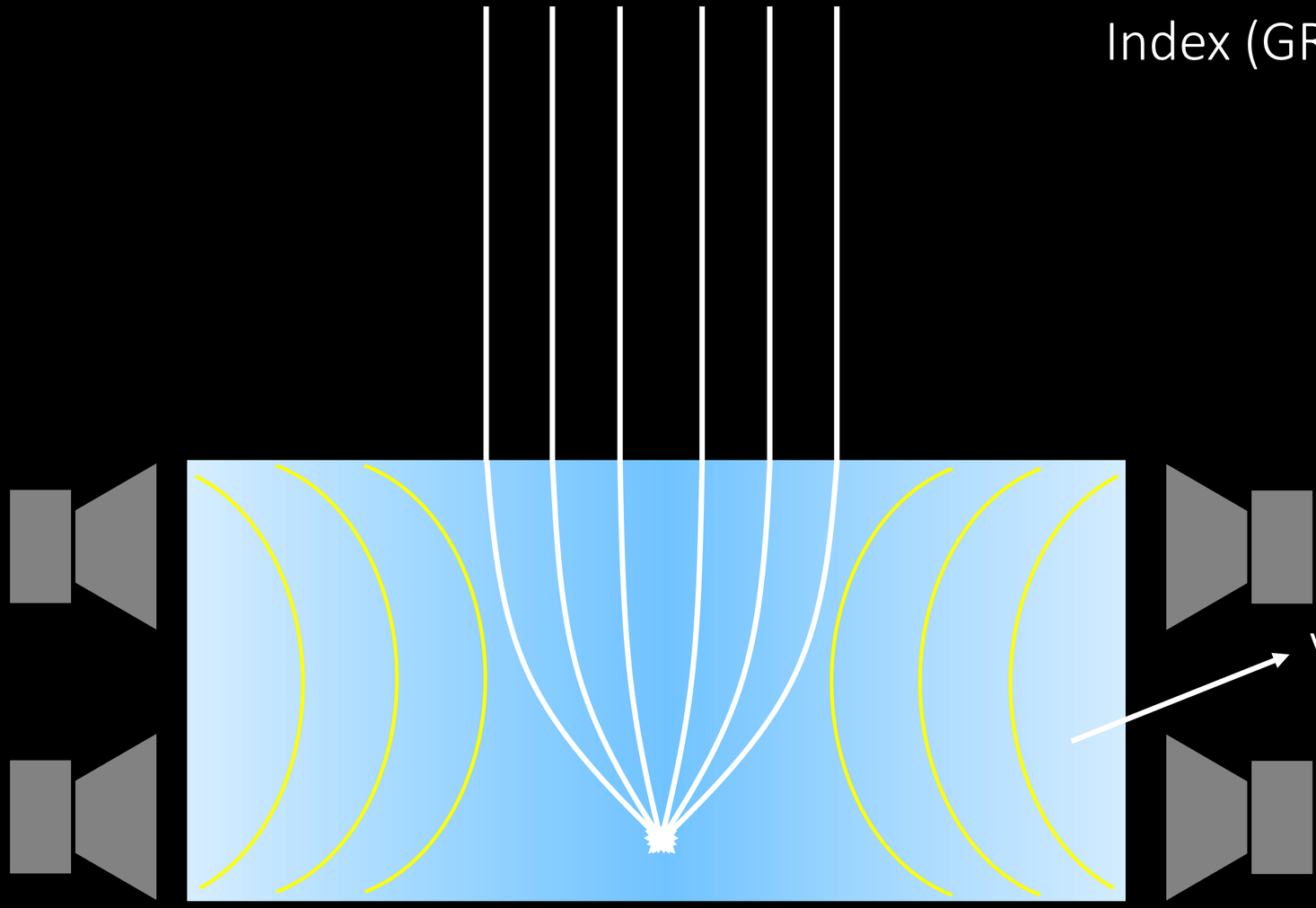
Gradient Refractive Index (GRIN) waveguide



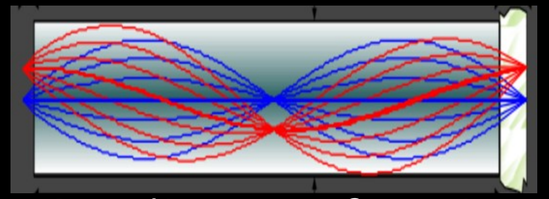
Focusing light inside tissue



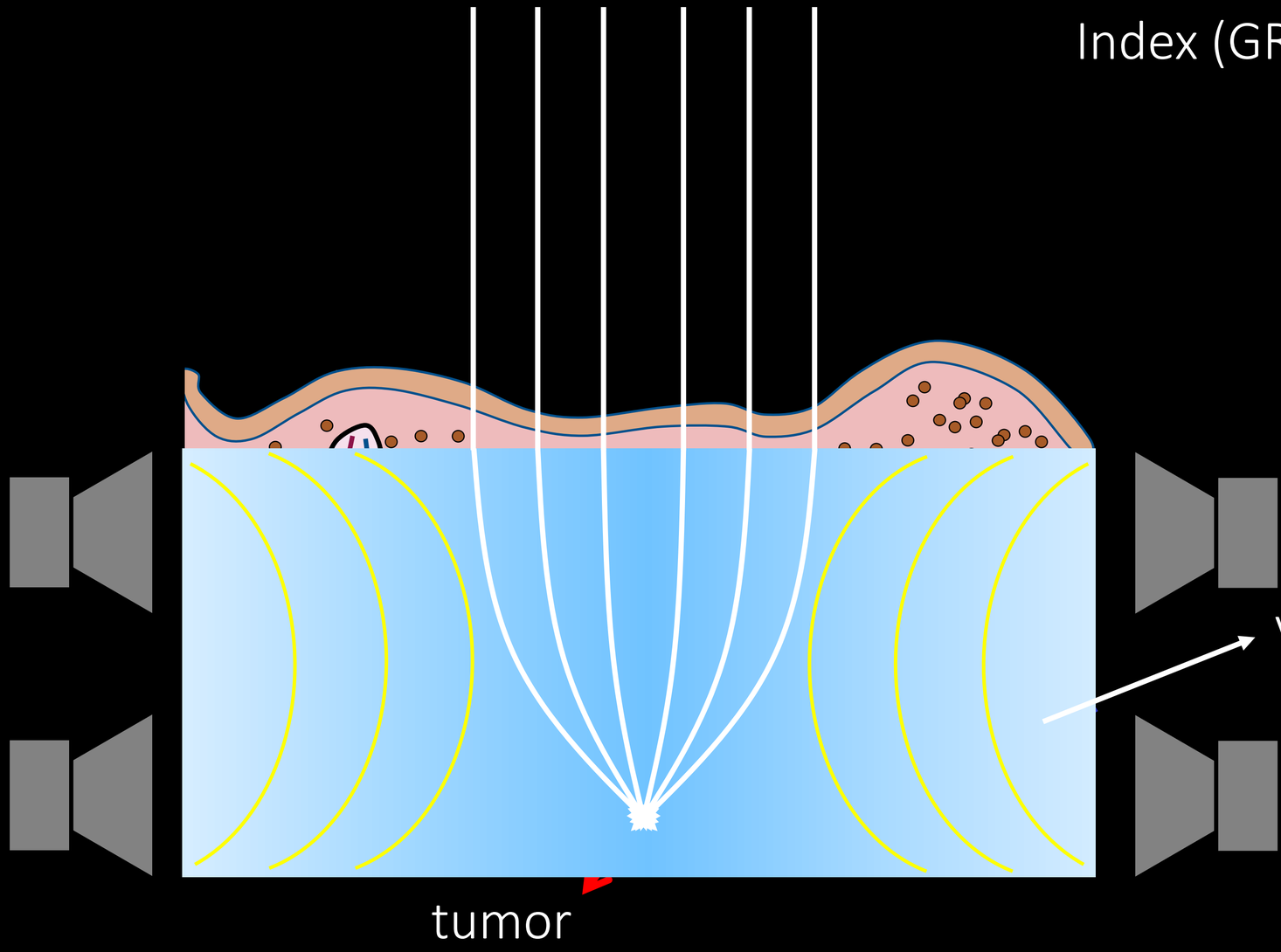
Gradient Refractive Index (GRIN) waveguide



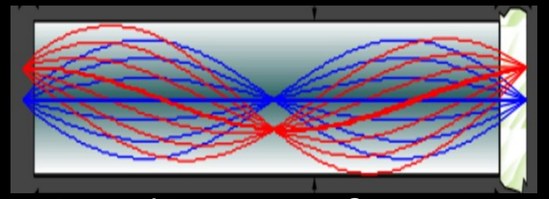
Focusing light inside tissue



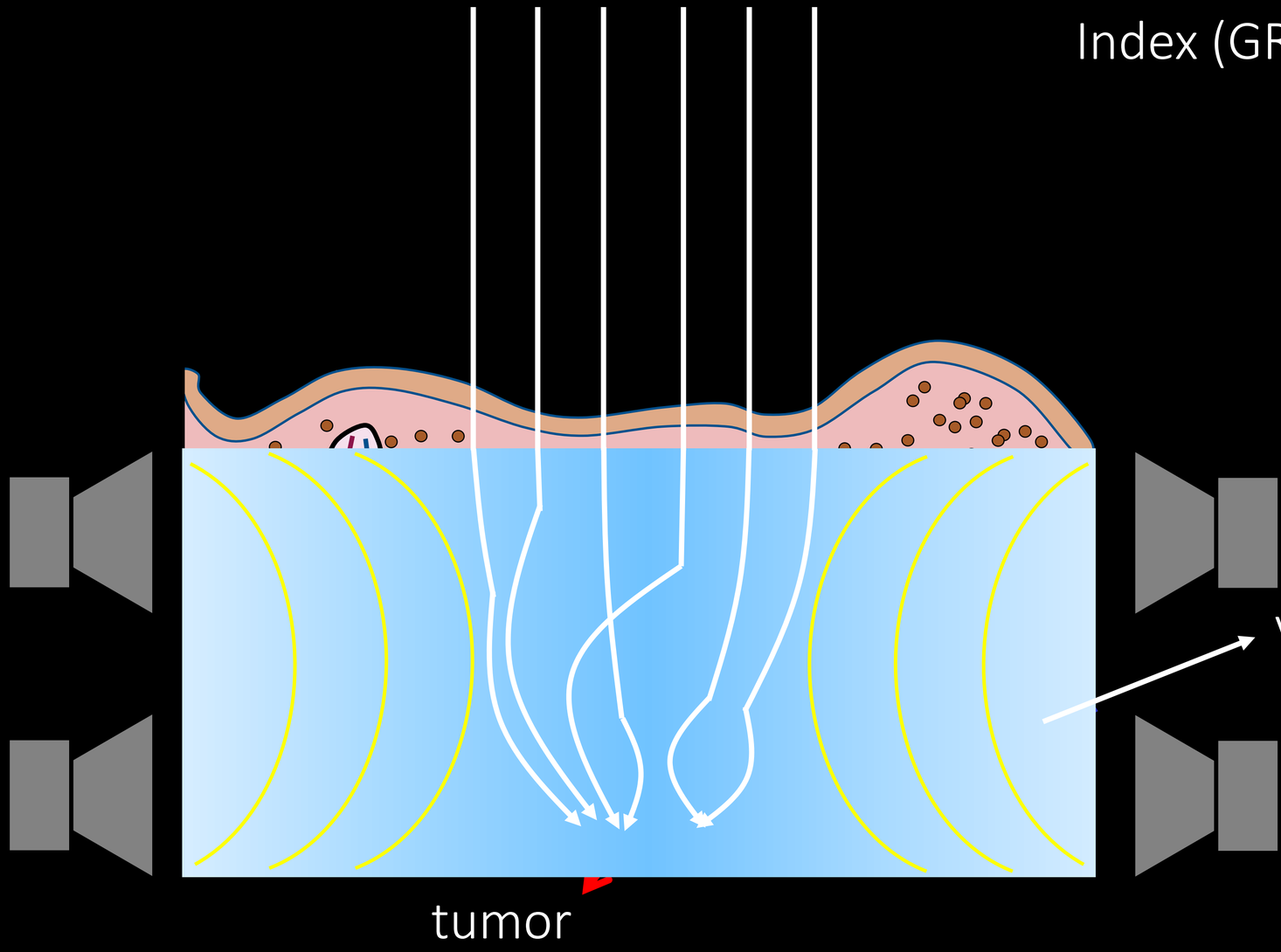
Gradient Refractive Index (GRIN) waveguide



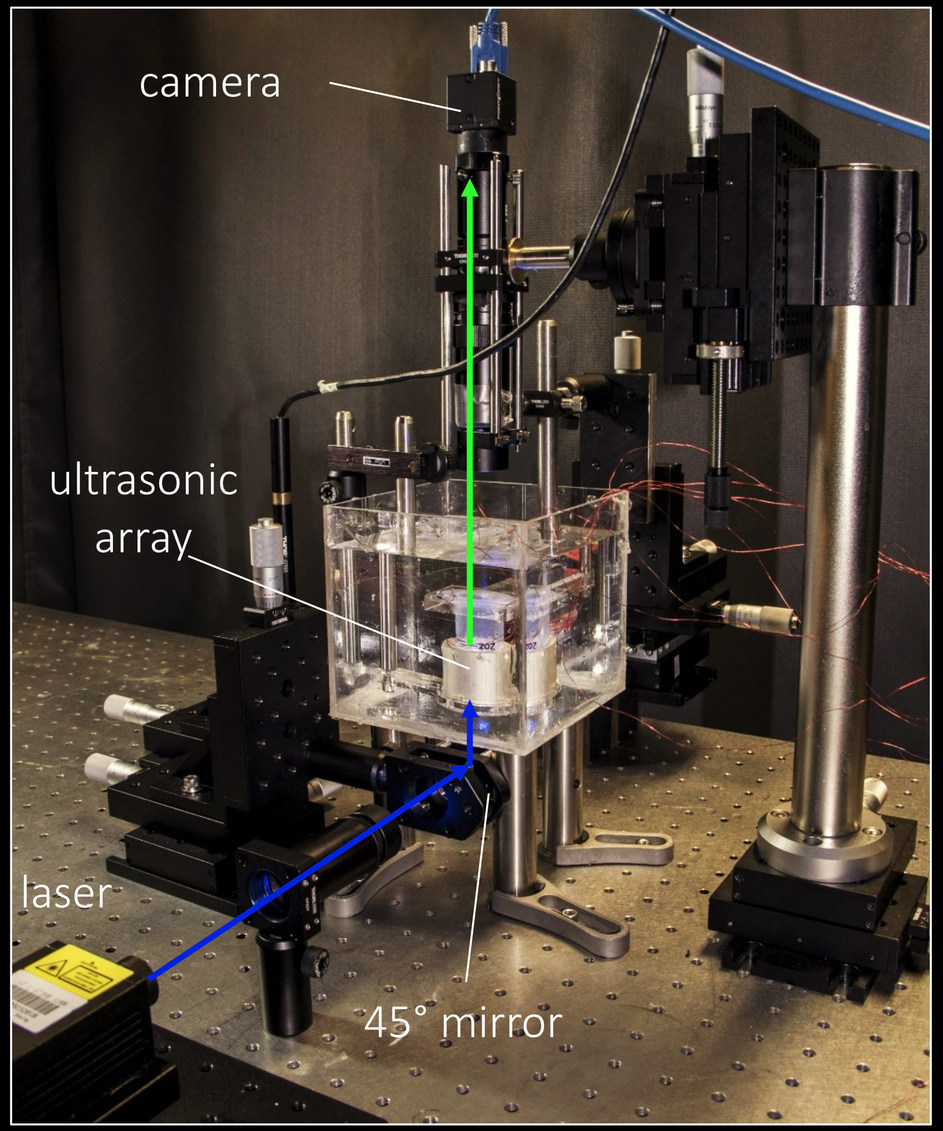
Focusing light inside tissue



Gradient Refractive Index (GRIN) waveguide

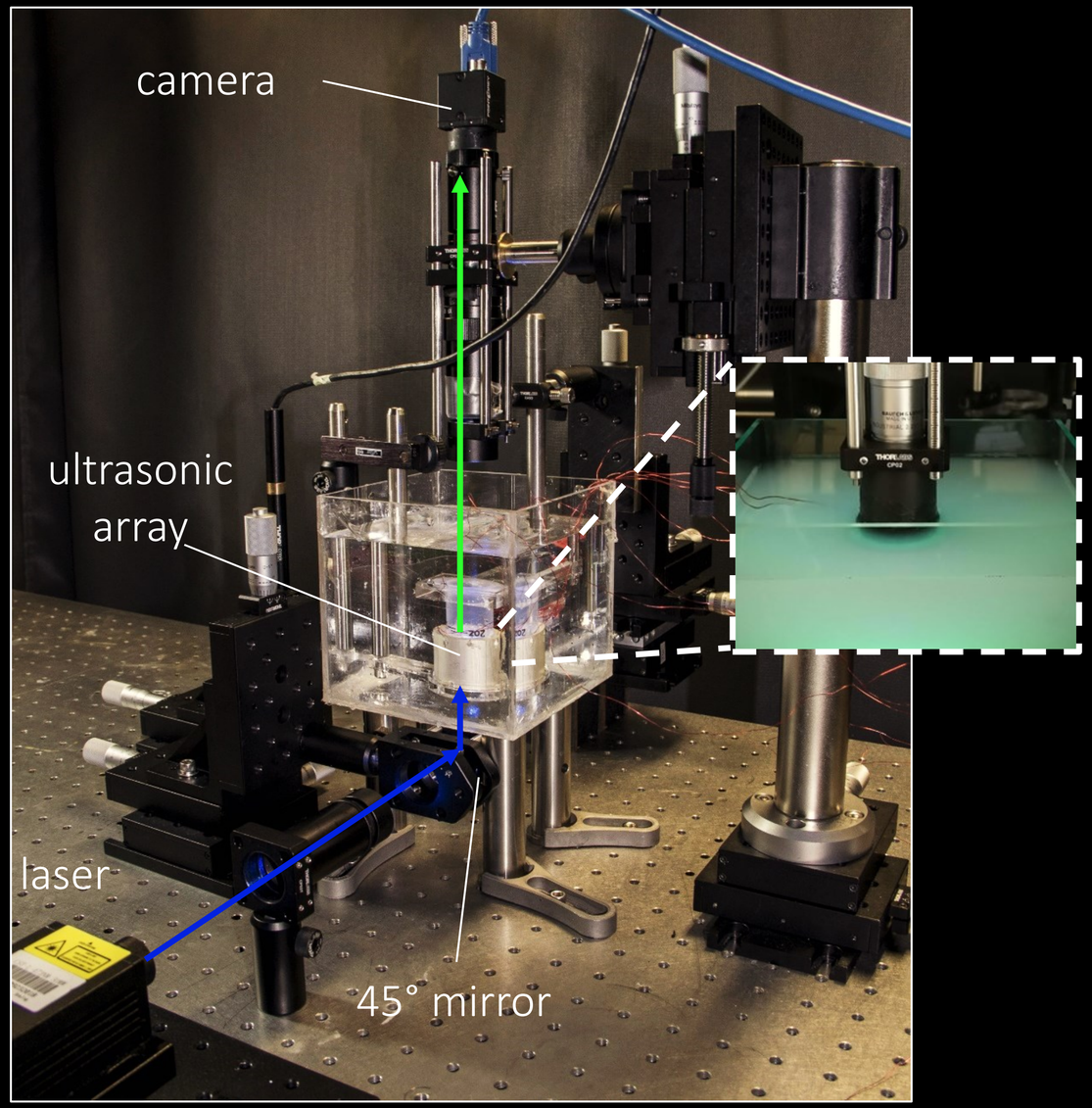


Ultrasonic light guiding inside tissue



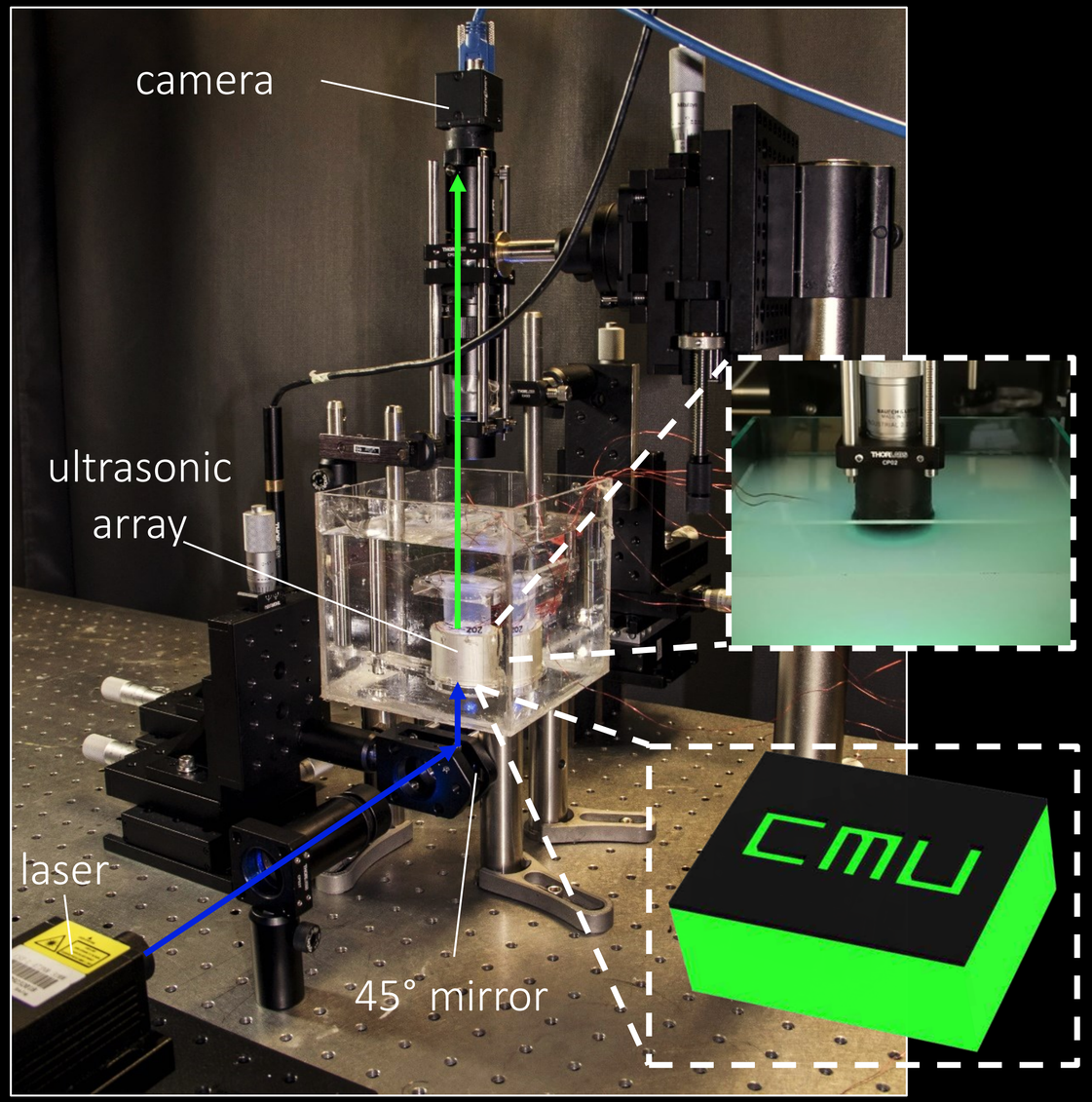
[Chamanzar et al., Nat. Comm. 2019]
[Karimi et al., Optics Express, 2019]
[Scopelliti et al., LSA, 2019]

Ultrasonic light guiding inside tissue



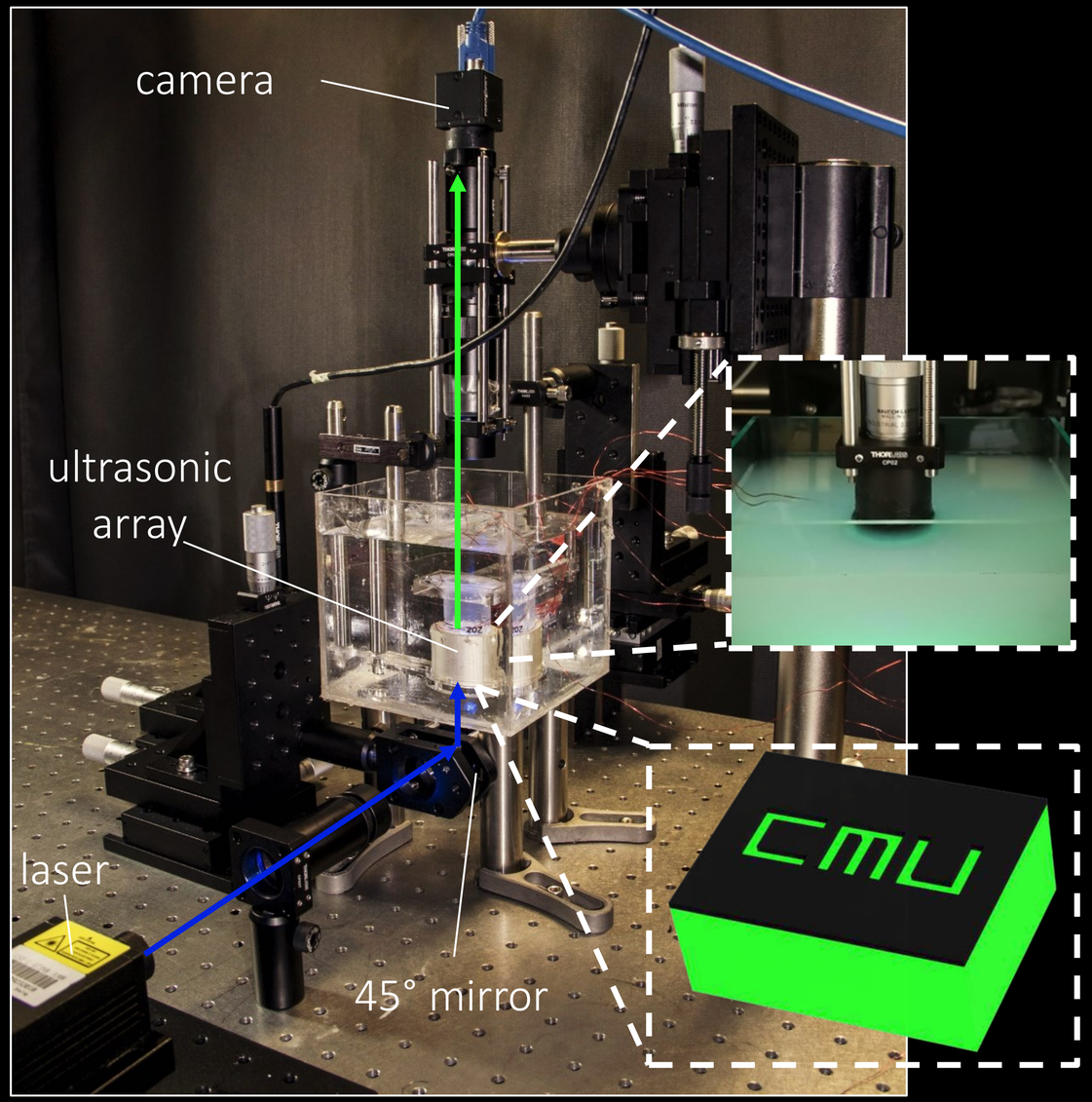
[Chamanzar et al., Nat. Comm. 2019]
[Karimi et al., Optics Express, 2019]
[Scopelliti et al., LSA, 2019]

Ultrasonic light guiding inside tissue

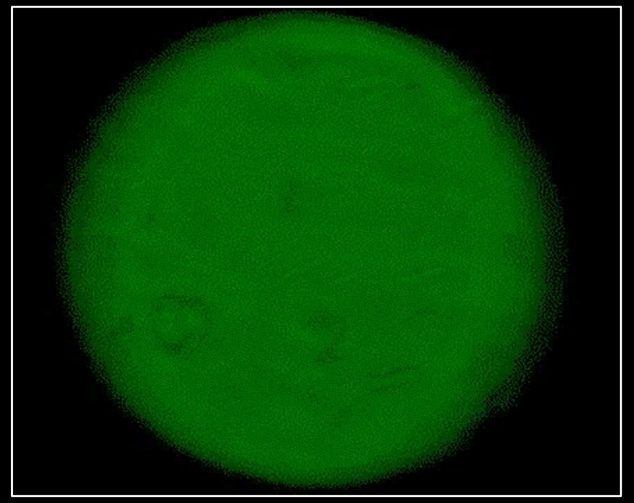


[Chamanzar et al., Nat. Comm. 2019]
[Karimi et al., Optics Express, 2019]
[Scopelliti et al., LSA, 2019]

Ultrasonic light guiding inside tissue

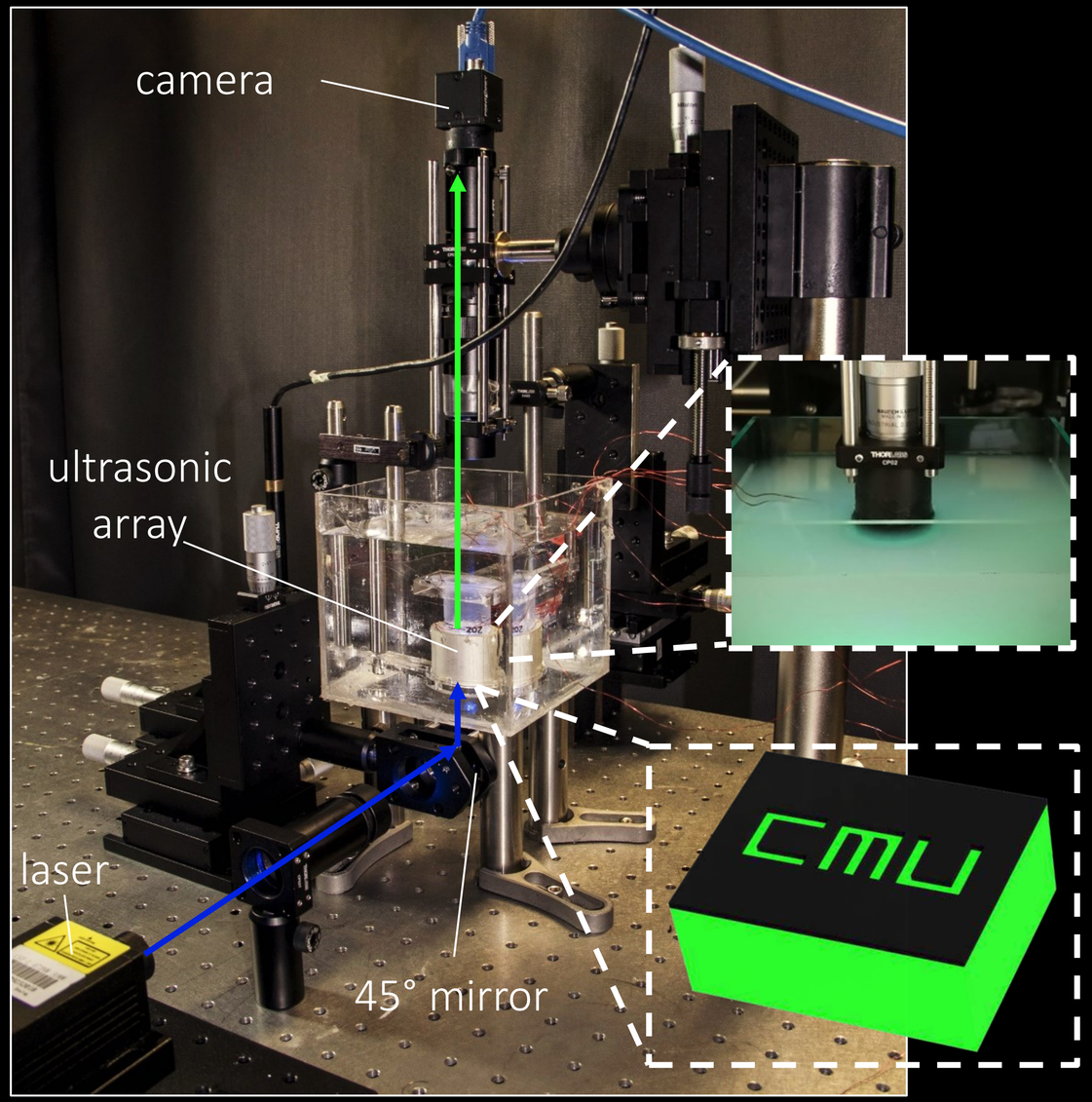


ultrasound off

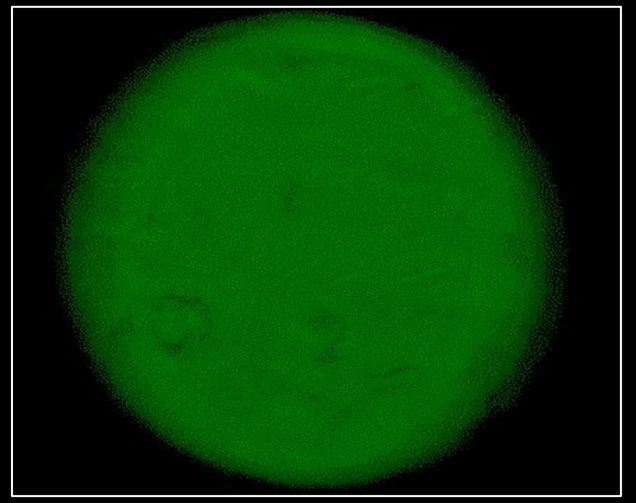


[Chamanzar et al., Nat. Comm. 2019]
[Karimi et al., Optics Express, 2019]
[Scopelliti et al., LSA, 2019]

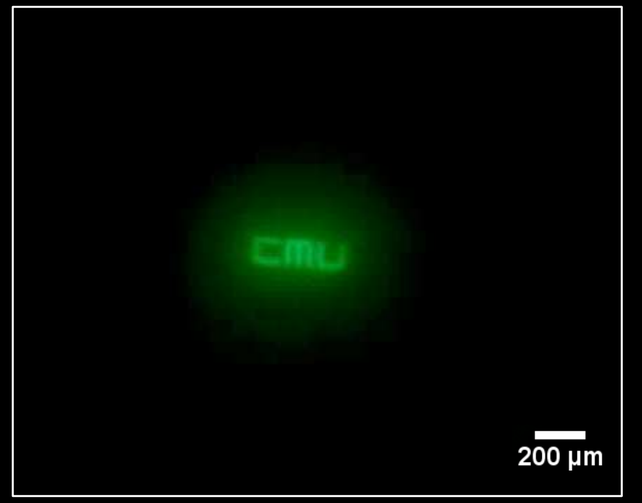
Ultrasonic light guiding inside tissue



ultrasound off

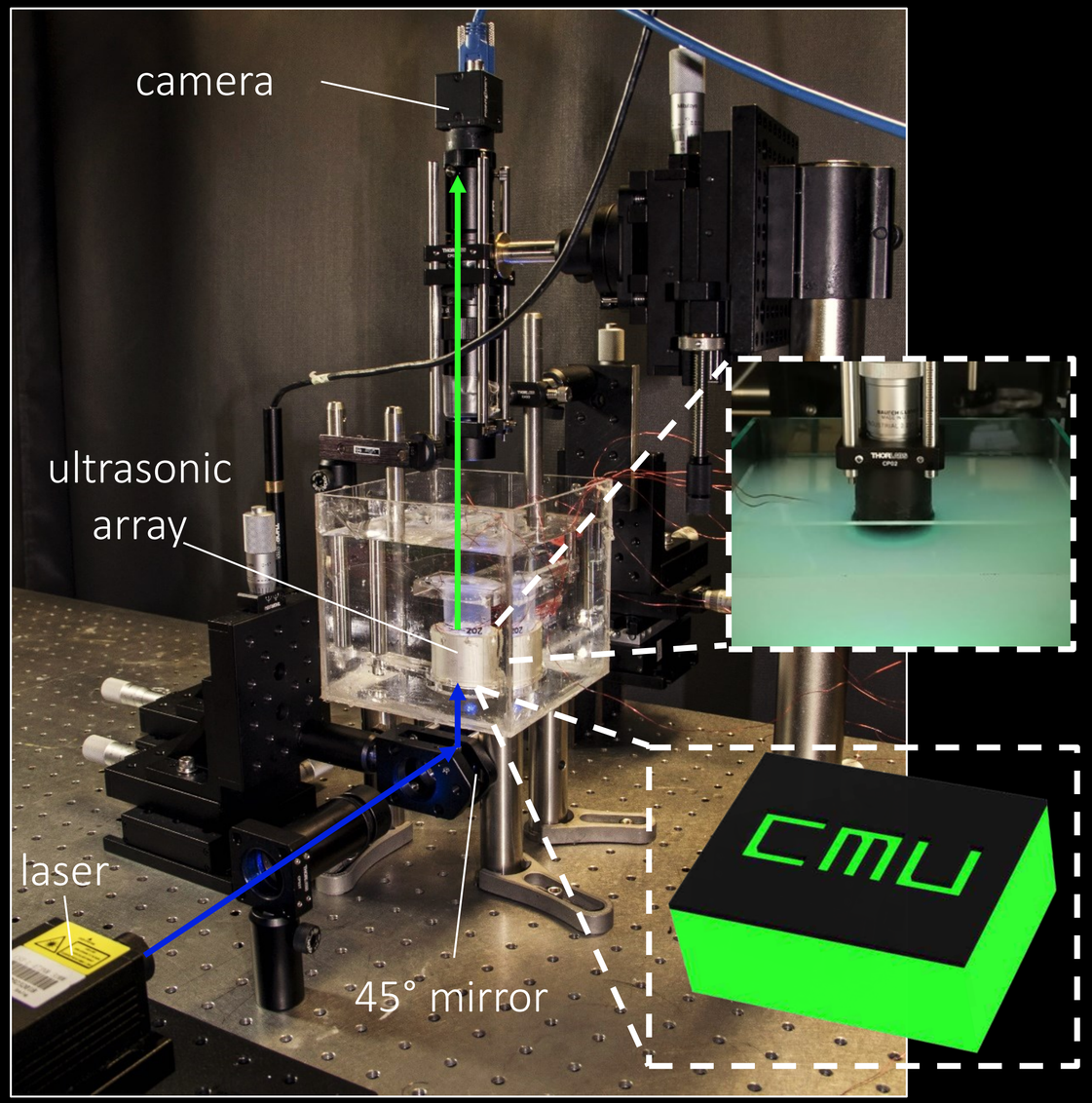


ultrasound on



[Chamanzar et al., Nat. Comm. 2019]
[Karimi et al., Optics Express, 2019]
[Scopelliti et al., LSA, 2019]

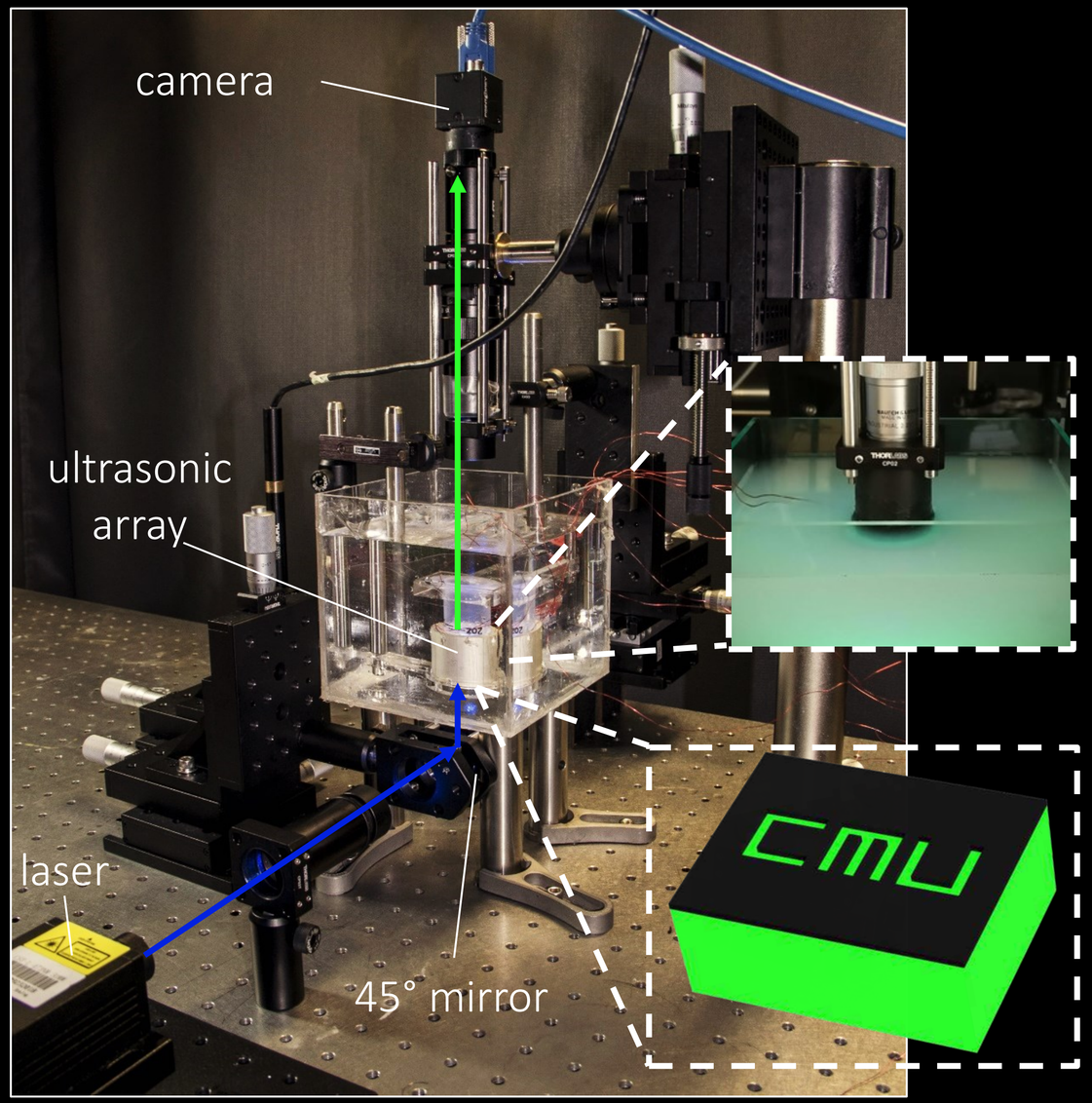
Ultrasonic light guiding inside tissue



High-dimensional, highly-non-linear design problem:

- ultrasound frequency
- ultrasound voltage
- placement of transducers
- shape of waveguides
- waveform shape
- and more…

Ultrasonic light guiding inside tissue

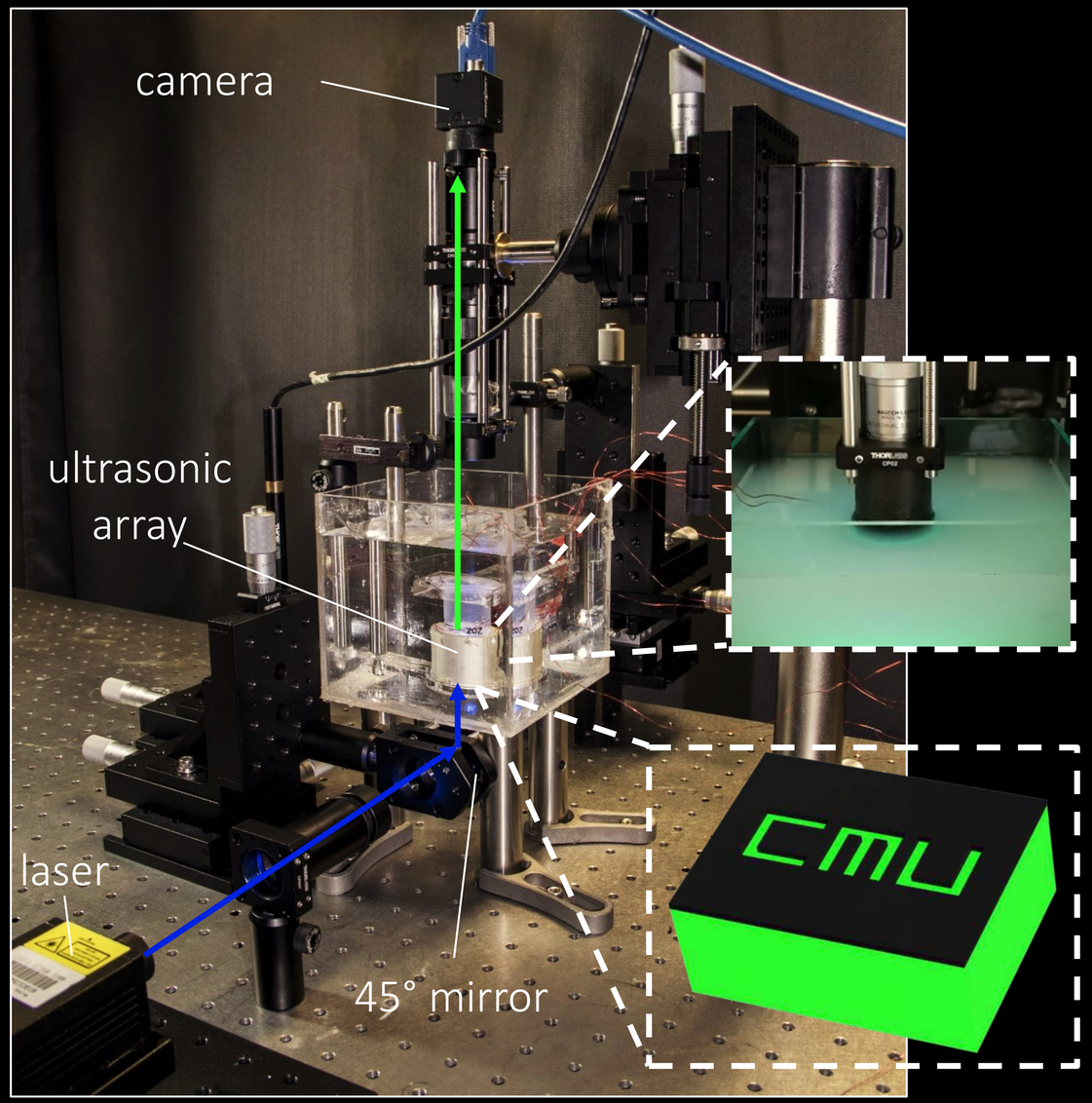


High-dimensional, highly-non-linear design problem:

- ultrasound frequency
- ultrasound voltage
- placement of transducers
- shape of waveguides
- waveform shape
- and more…

Efficiently explore using rendering

Ultrasonic light guiding inside tissue



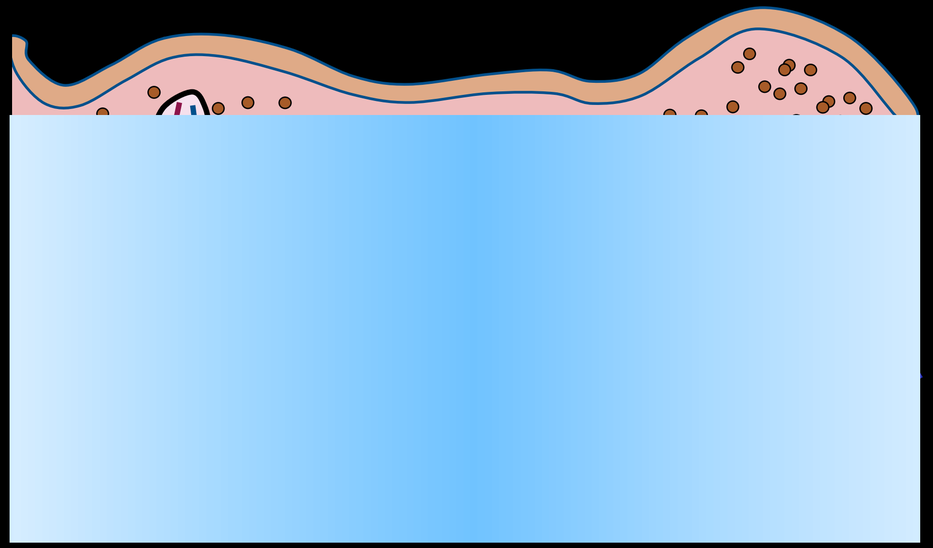
High-dimensional, highly-non-linear design problem:

- ultrasound frequency
- ultrasound voltage
- placement of transducers
- shape of waveguides
- waveform shape
- and more…

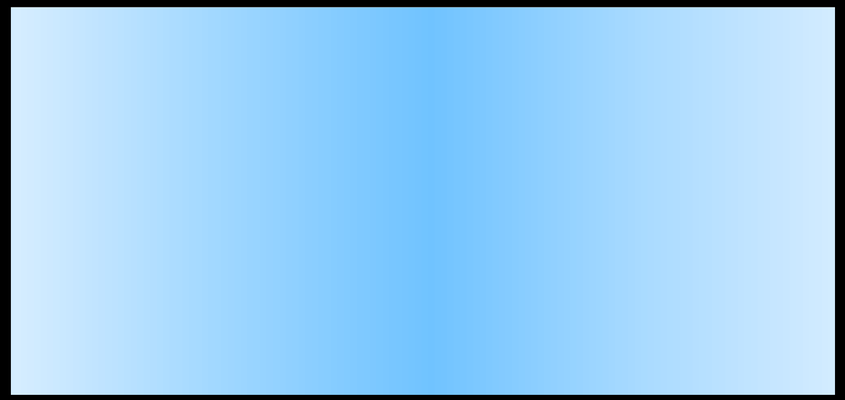
Efficiently explore using rendering

Build first rendering algorithm

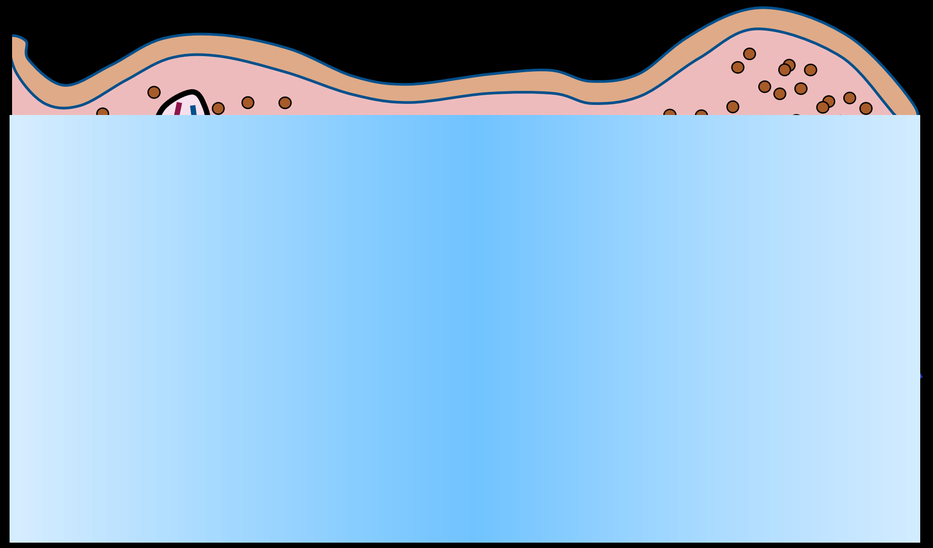
Rendering continuous refraction and scattering



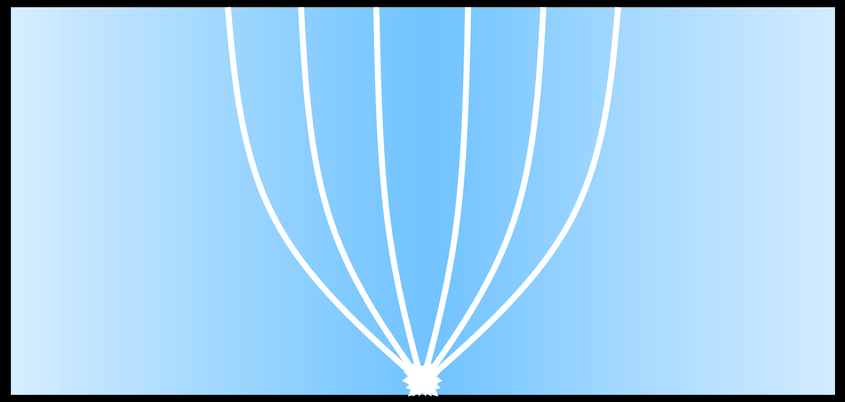
Rendering continuous refraction and scattering



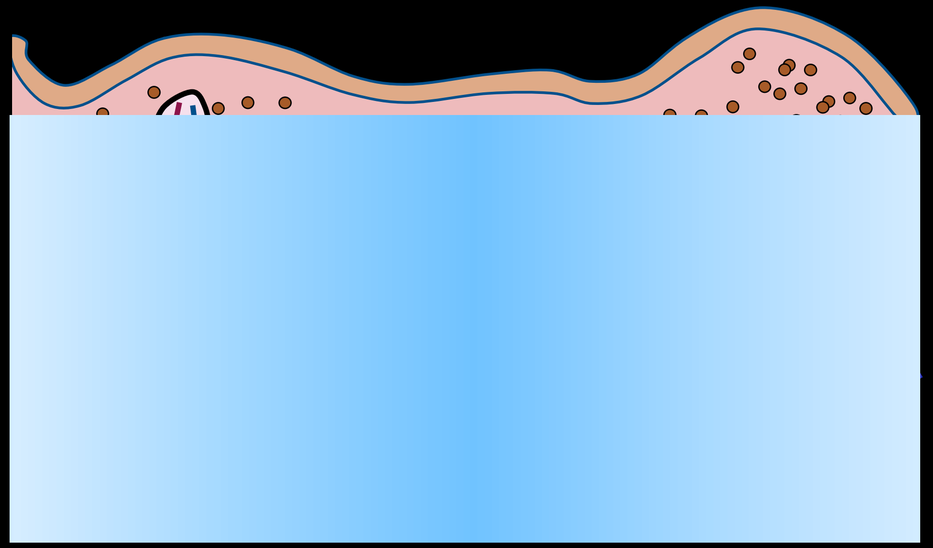
continuous refraction



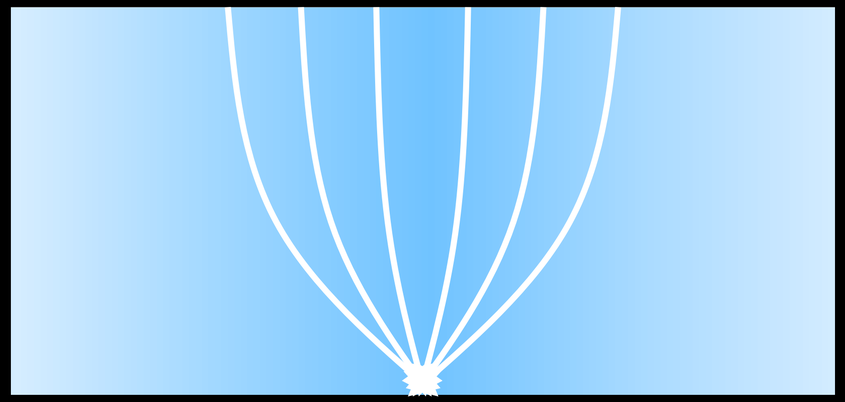
Rendering continuous refraction and scattering



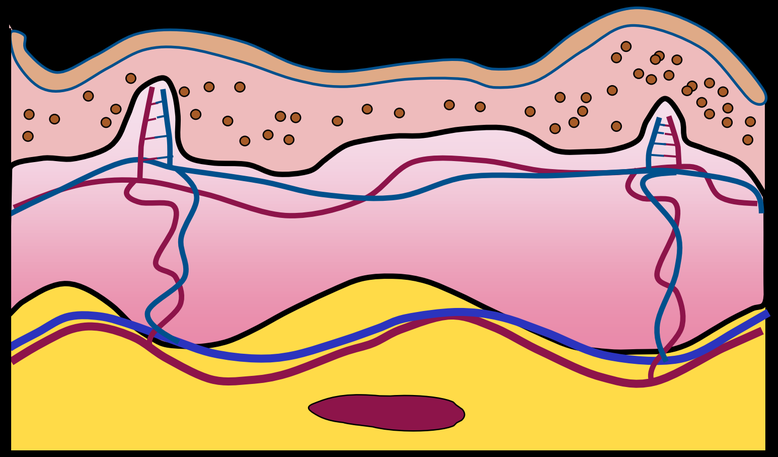
continuous refraction

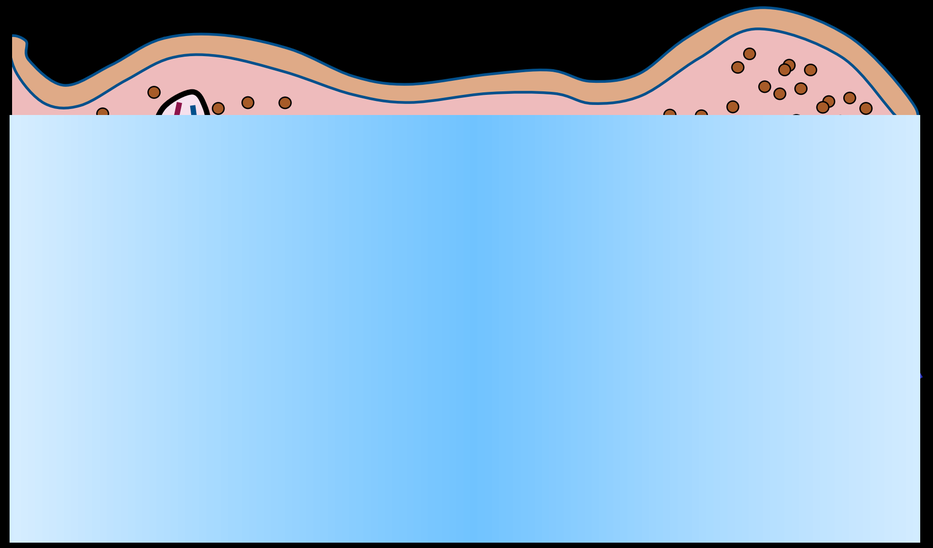


Rendering continuous refraction and scattering

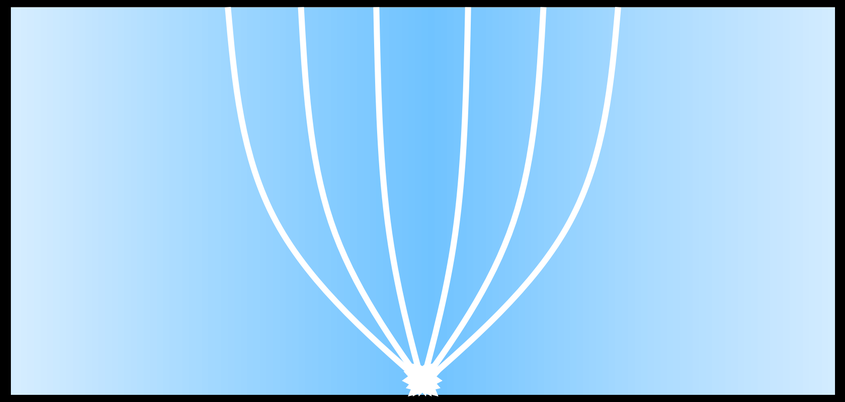


continuous refraction





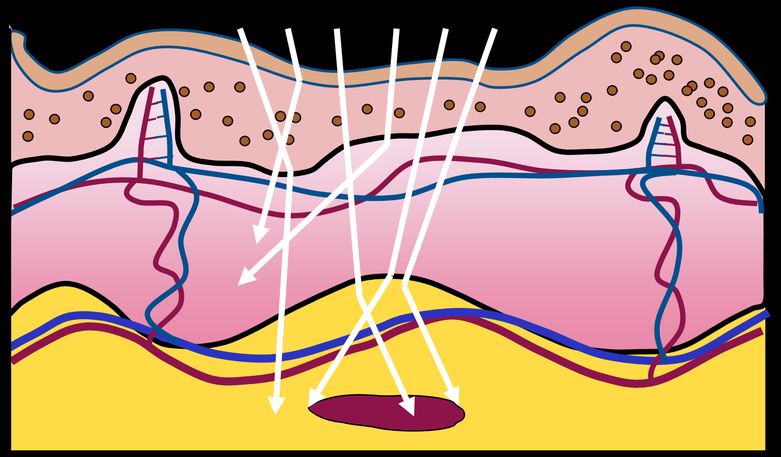
Rendering continuous refraction and scattering

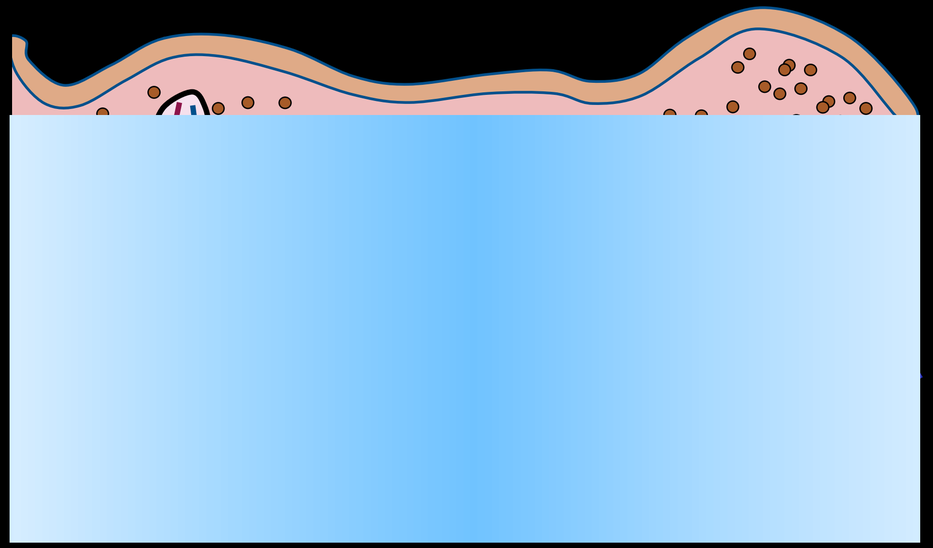


continuous refraction

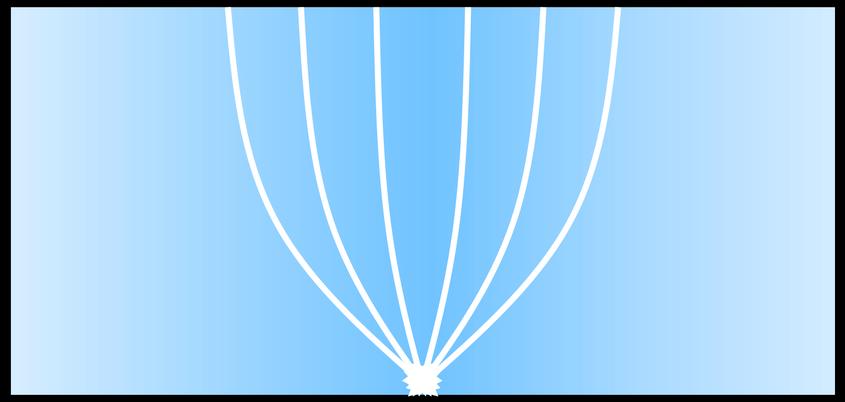
+

scattering





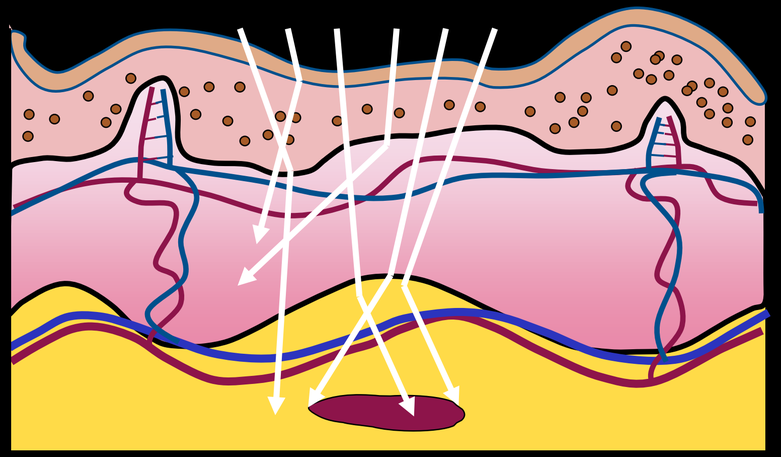
Rendering continuous refraction and scattering



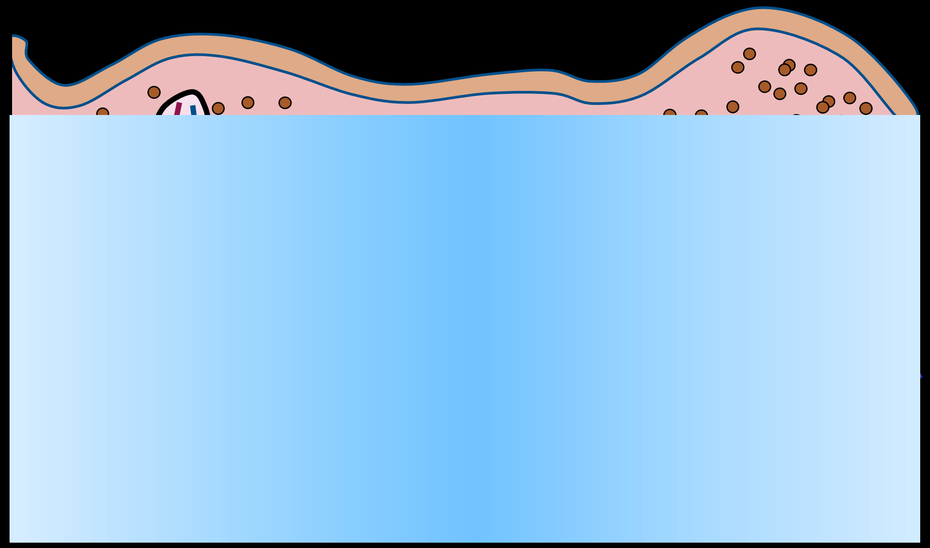
continuous refraction

+

scattering



[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

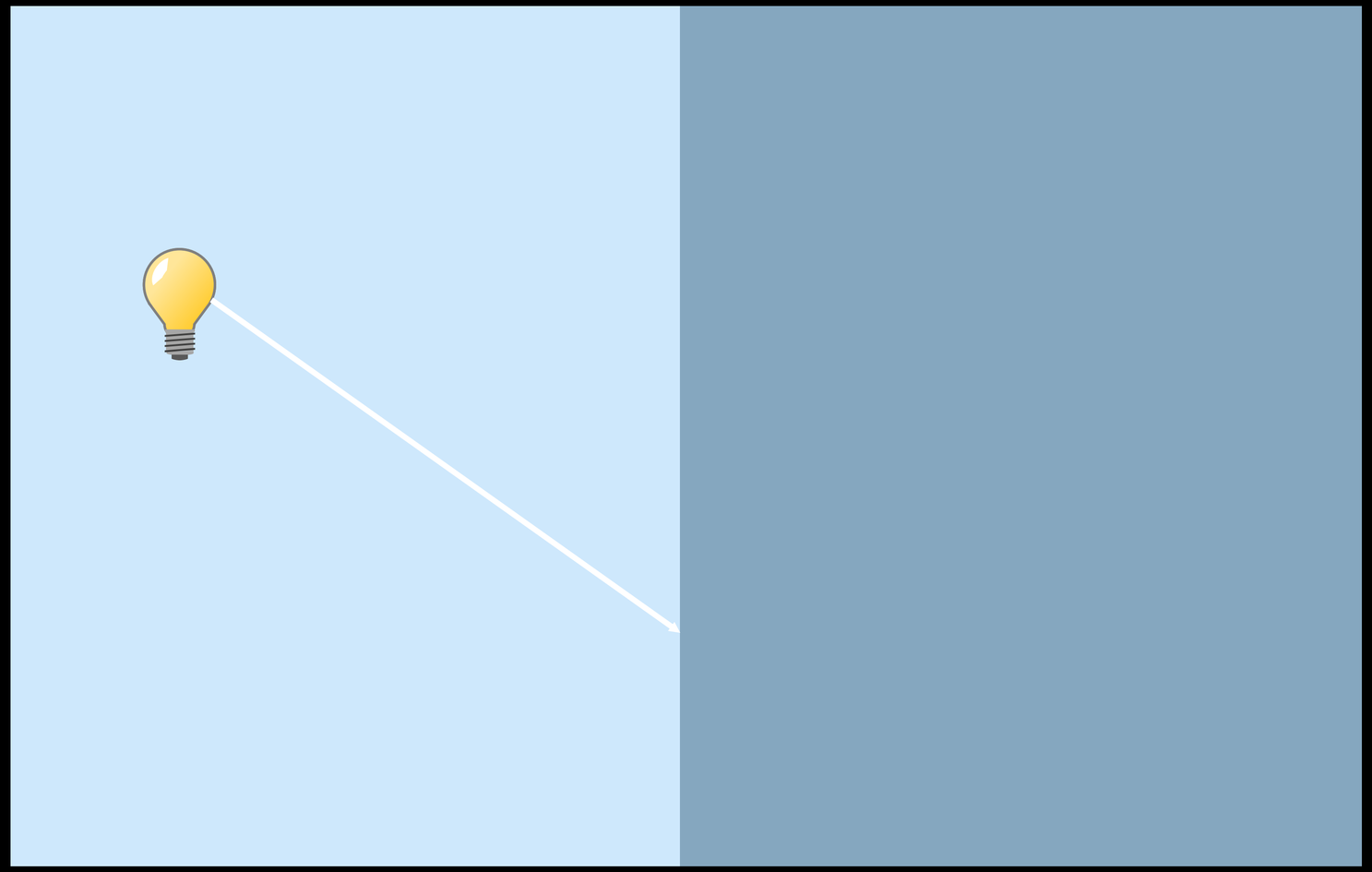


[Chandrasekhar, book, 1960]
[Lenoble, book, 1985]
[Lafortune and Willems, 1996]
[Cammarano and Jensen, 2002]
[Guttirez et al., Com. and Graph. 2006]
[Jarosz et al., Comp. Graph. forum, 2008]
[Jakob et al. ,ToG, 2010]
[Jarosz et al., ToG, 2011]
[Pediredla et al., JBO, 2016]
[Novak et al., Comp. Graph. forum, 2018]
[Bitterli et al., ToG, 2018]

Rendering continuous refraction ~~and scattering~~

[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

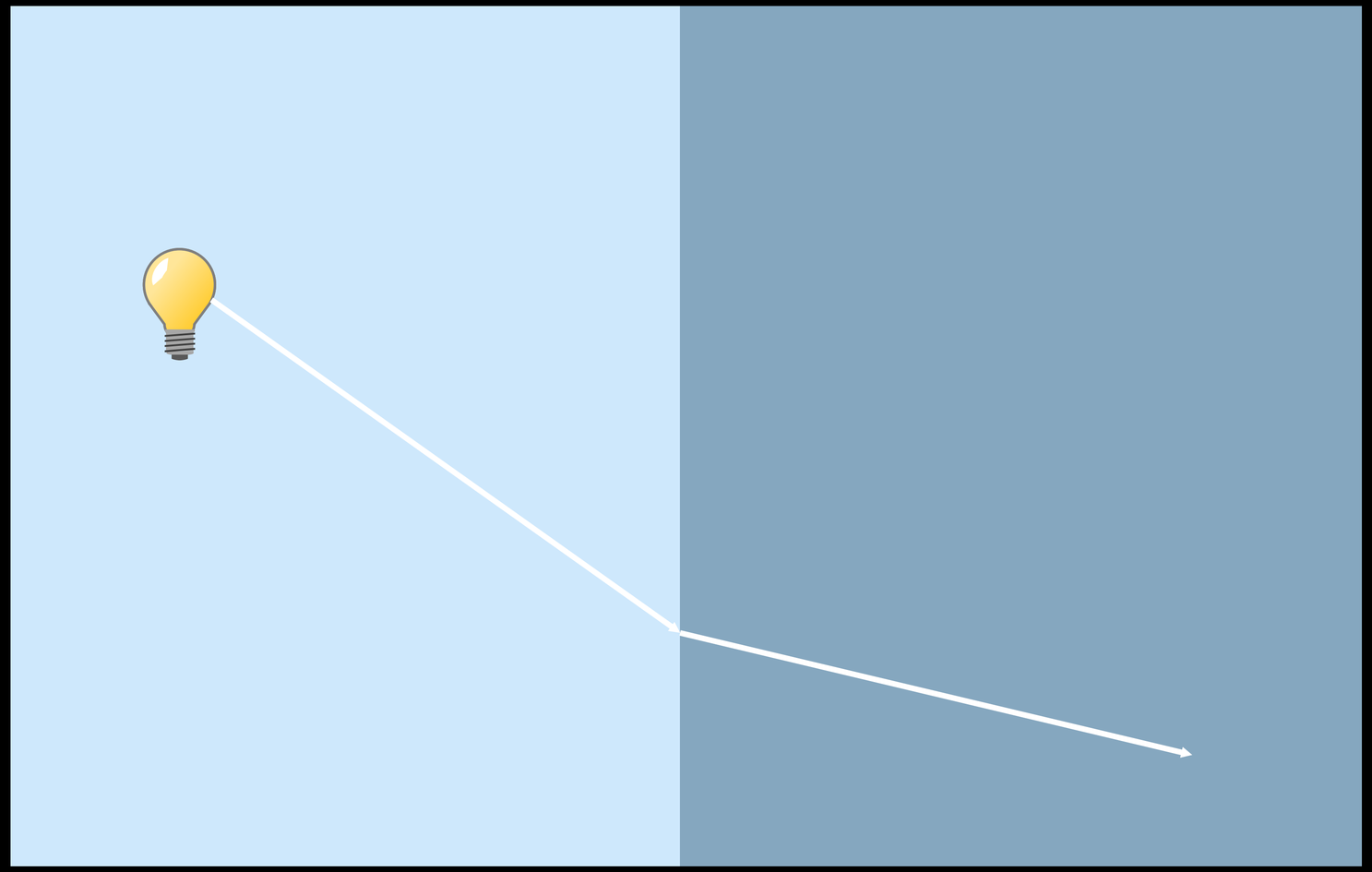
Rendering continuous refraction ~~and scattering~~



[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

Rendering continuous refraction ~~and scattering~~

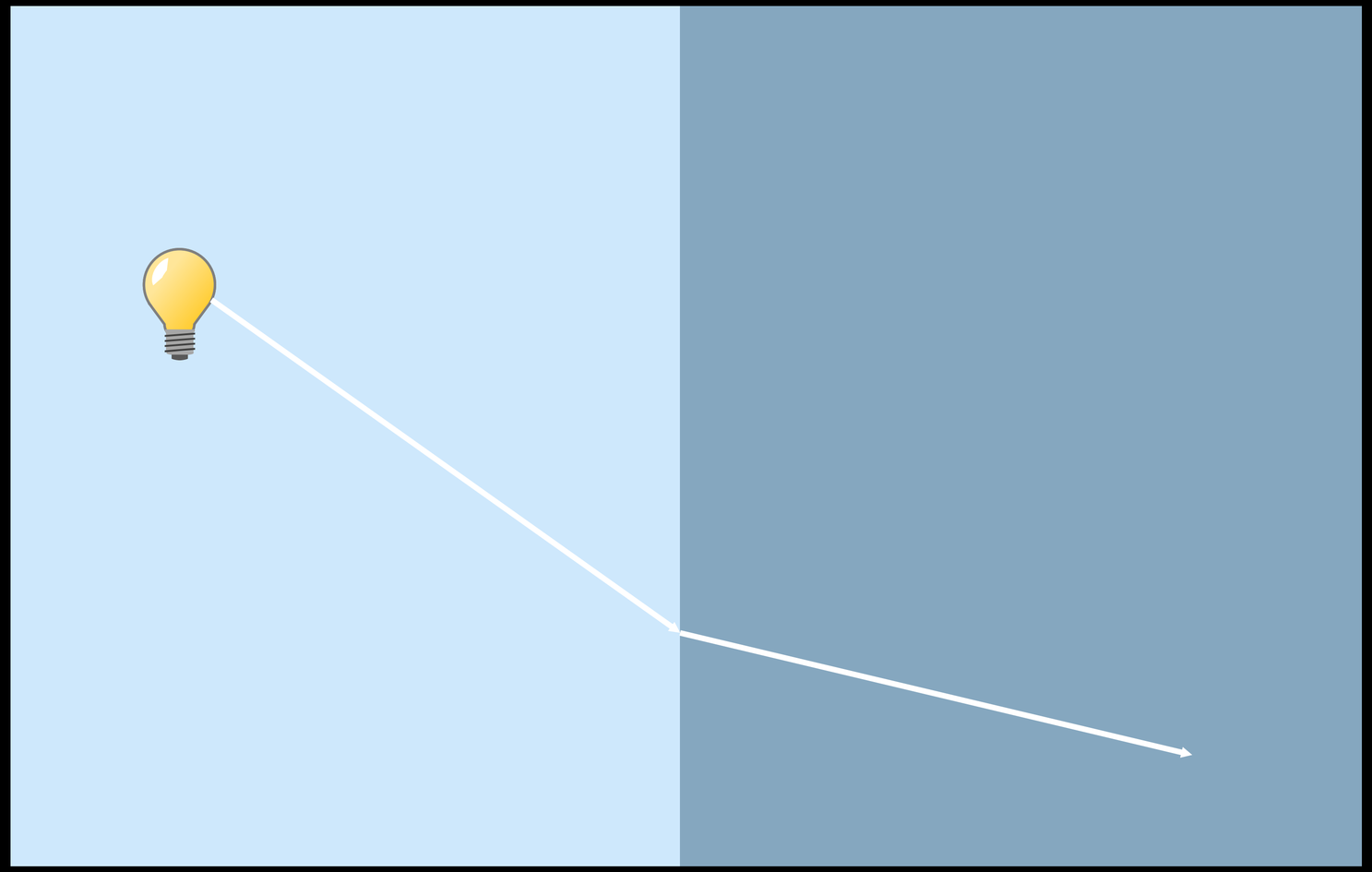
[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]



Rendering continuous refraction ~~and scattering~~

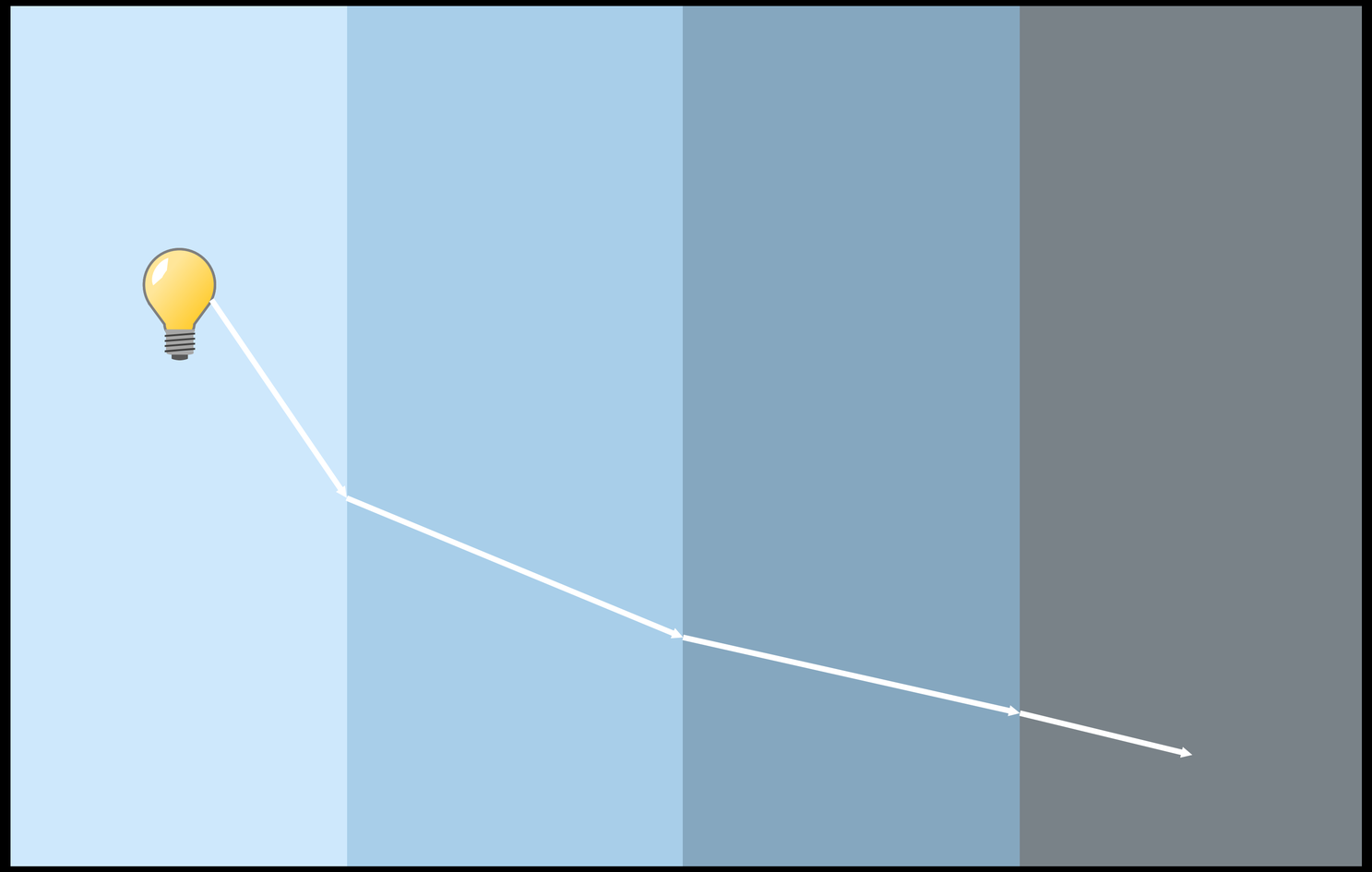
Snell's law

[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]



Rendering continuous refraction ~~and scattering~~

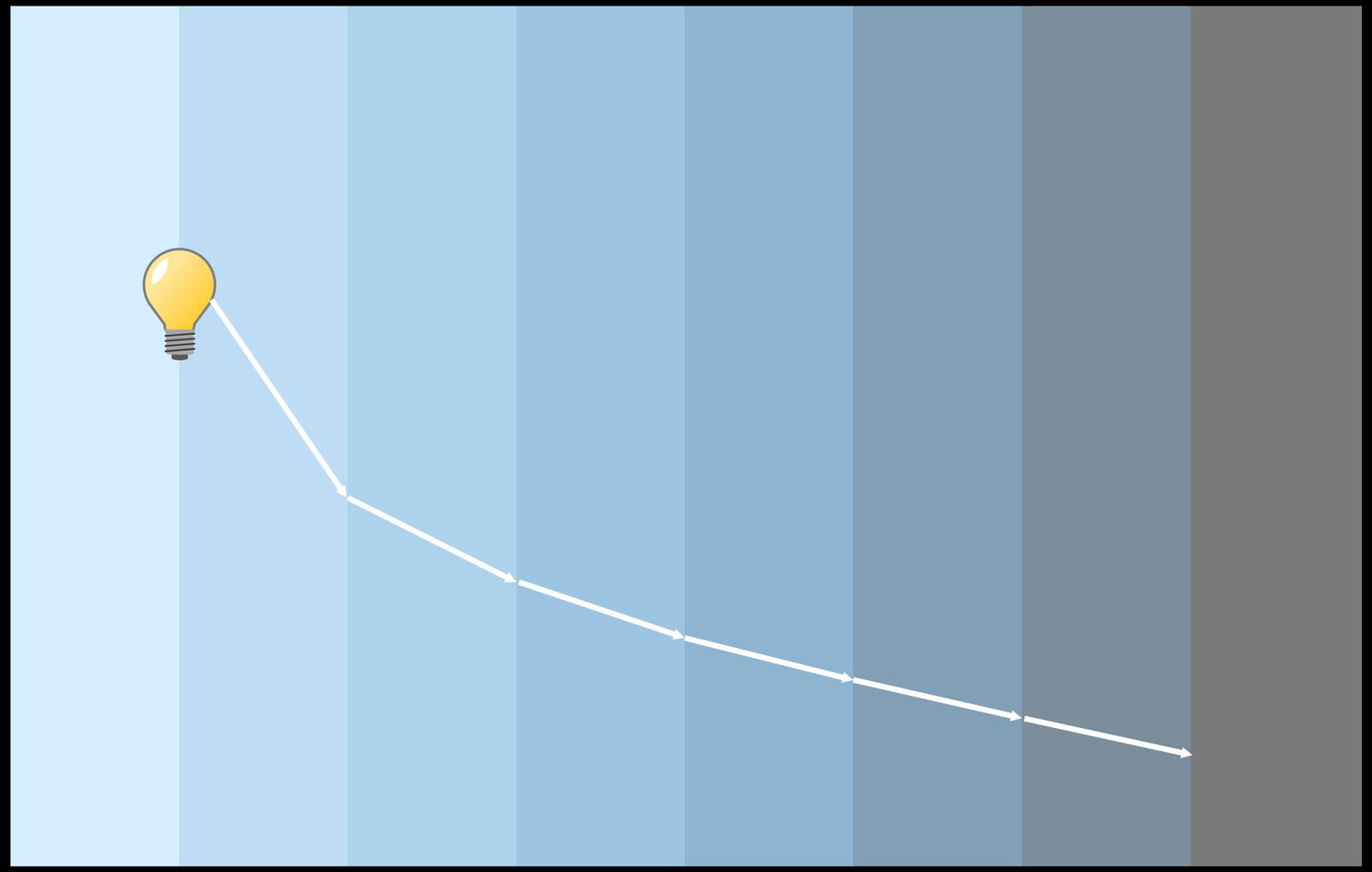
Snell's law



[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

Rendering continuous refraction ~~and scattering~~

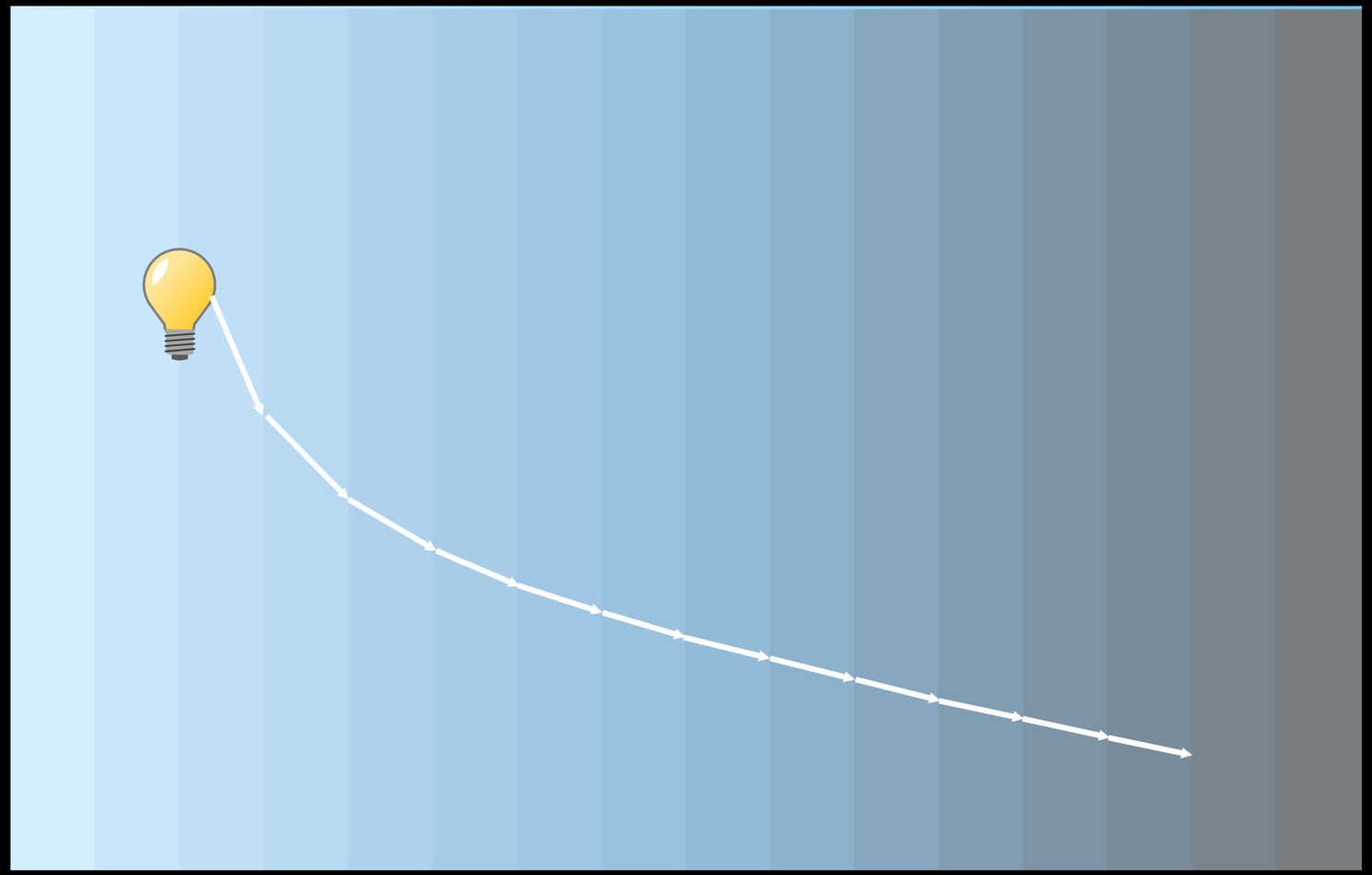
Snell's law



[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

Rendering continuous refraction ~~and scattering~~

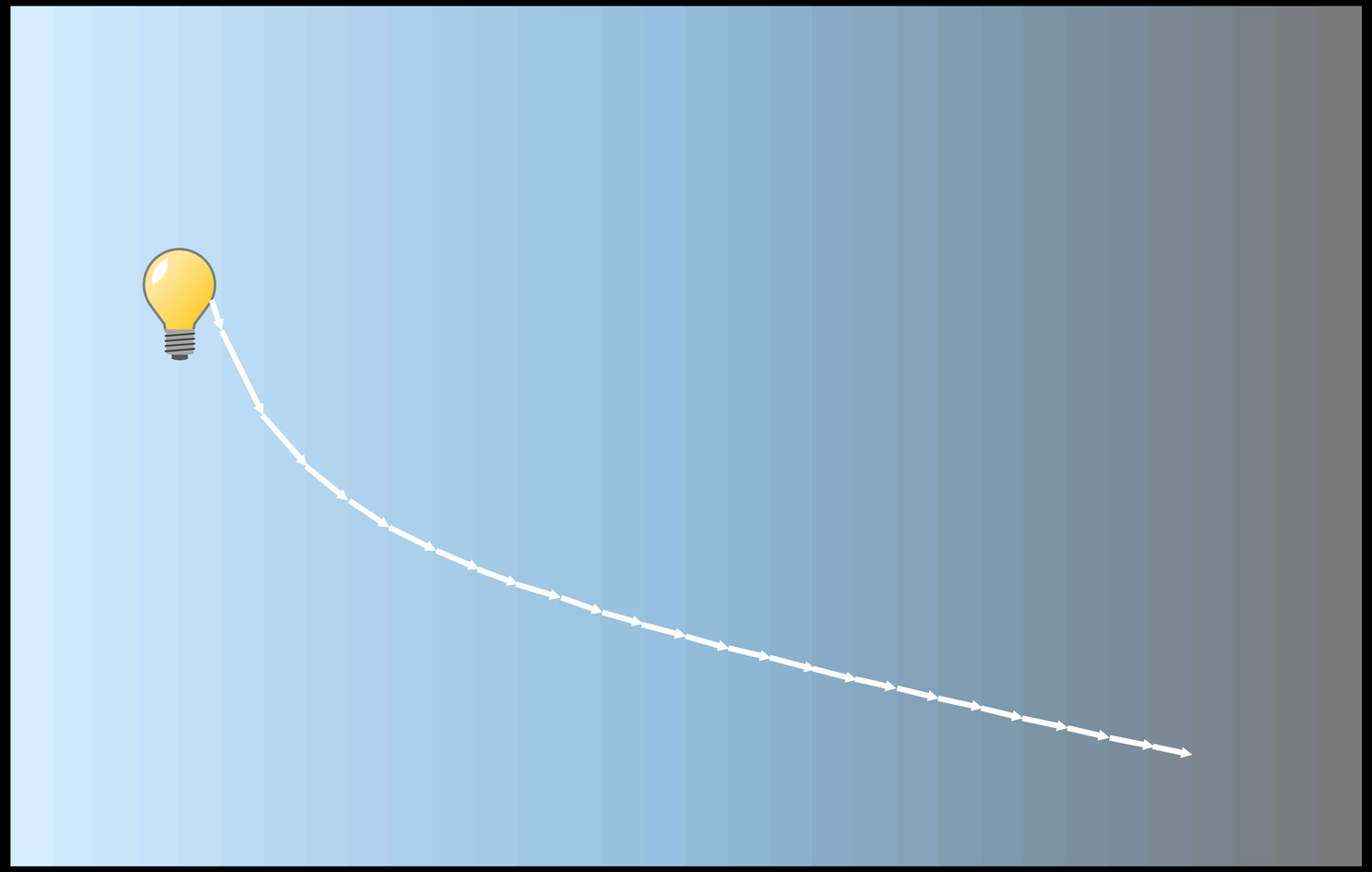
Snell's law



[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

Rendering continuous refraction ~~and scattering~~

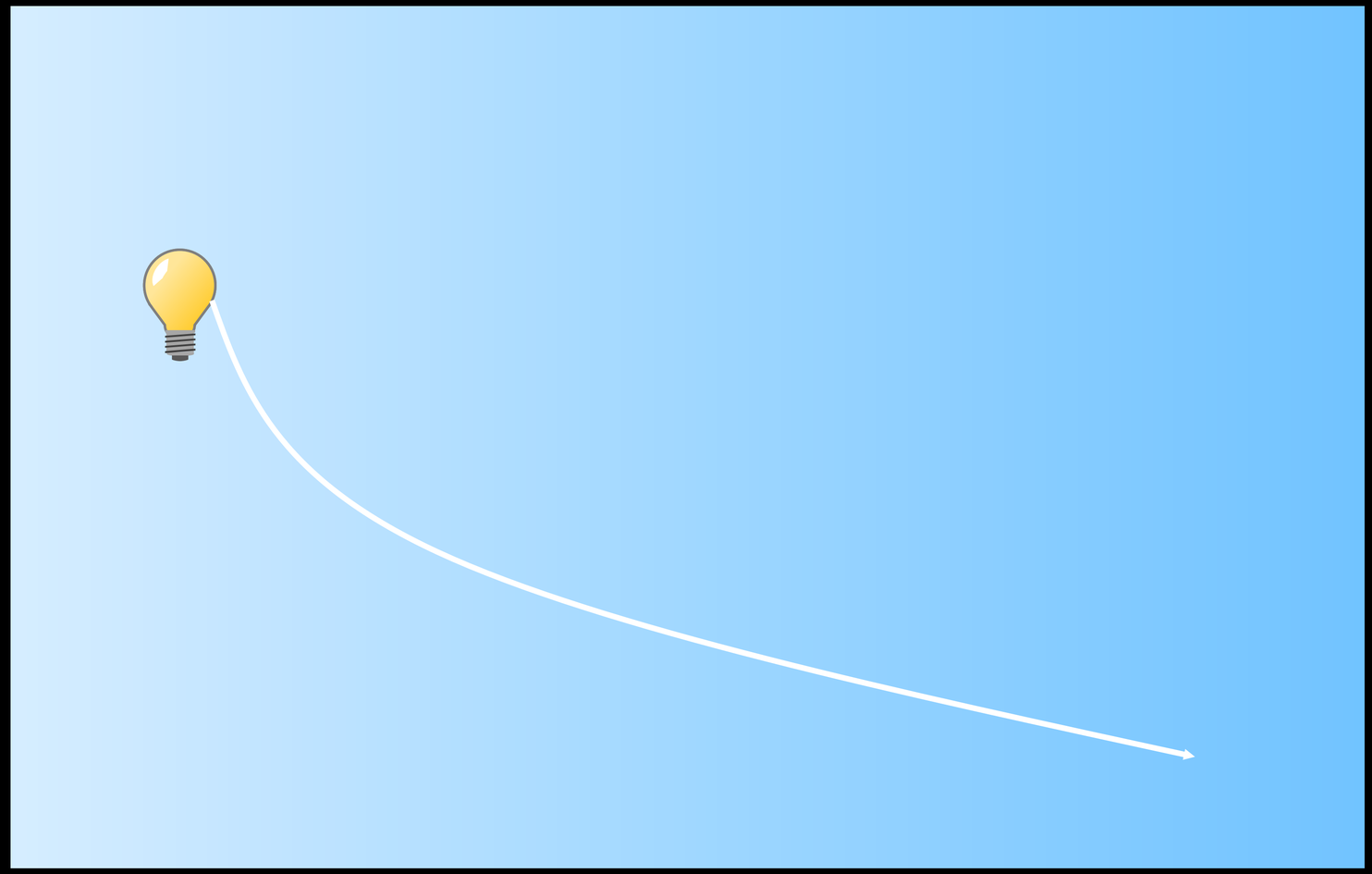
Snell's law



[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

Rendering continuous refraction ~~and scattering~~

refractive ray tracing

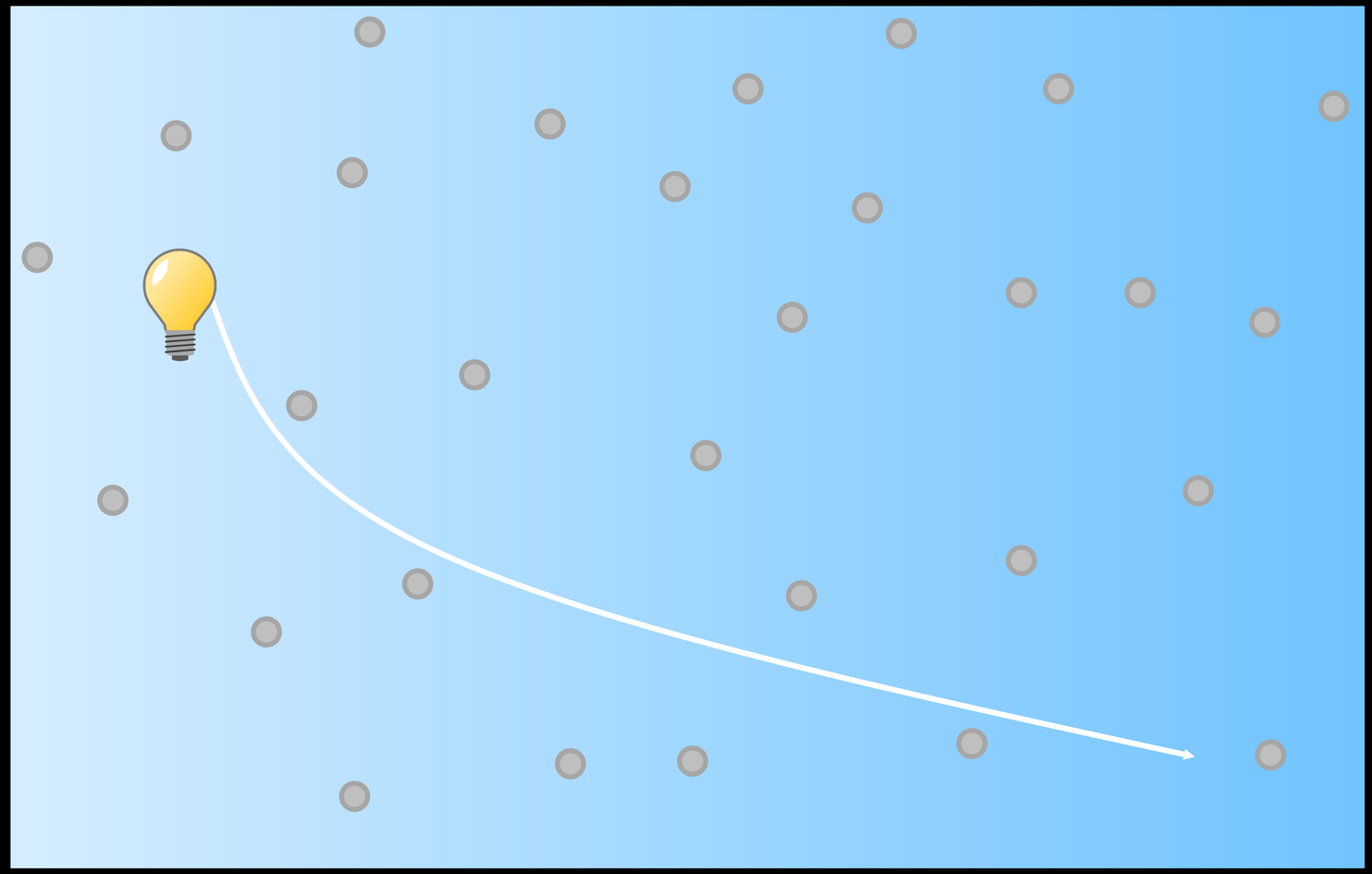


[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]

Rendering continuous refraction and scattering

refractive ray tracing

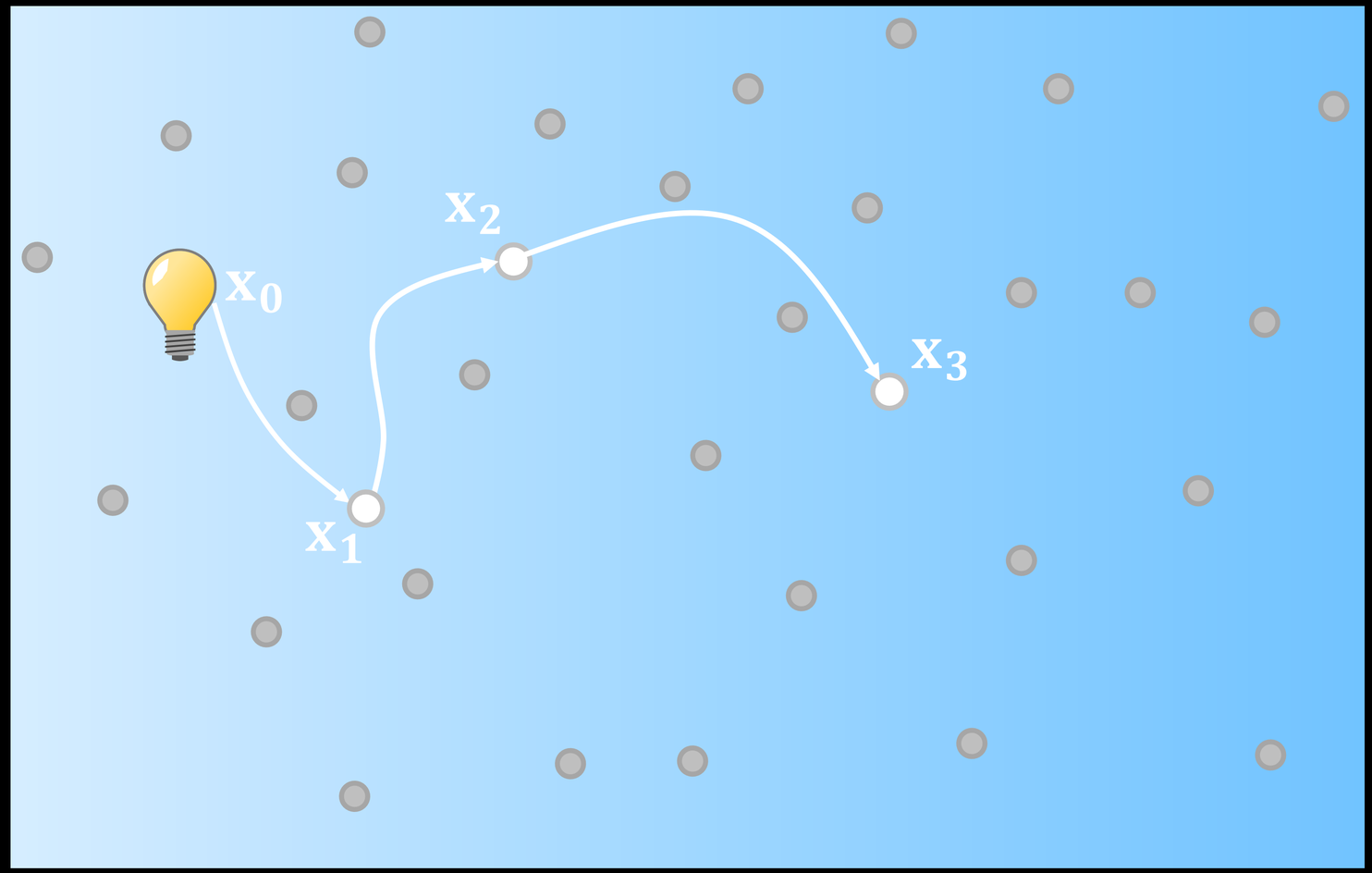
[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]



Rendering continuous refraction and scattering

refractive ray tracing

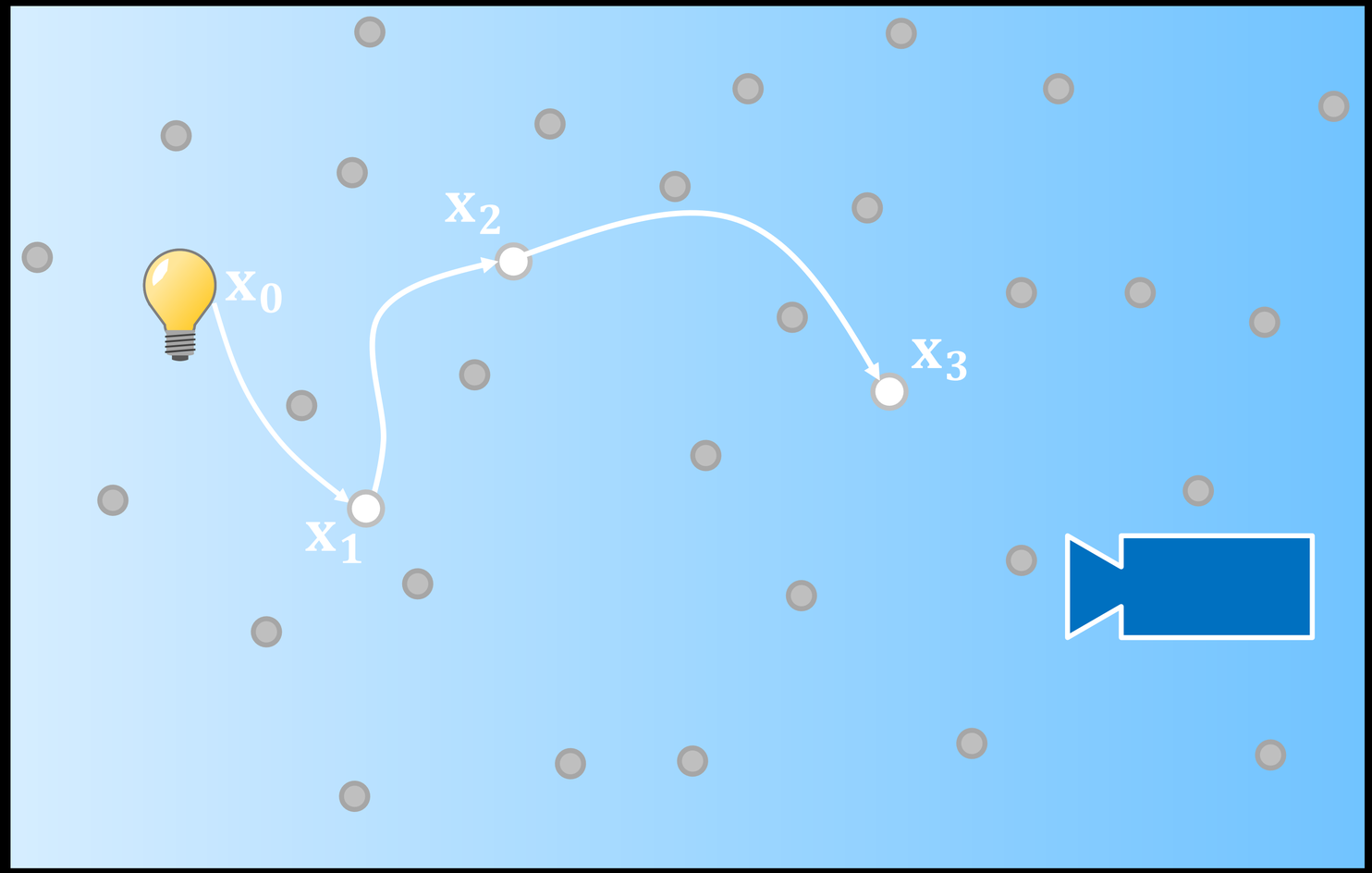
[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]



Rendering continuous refraction and scattering

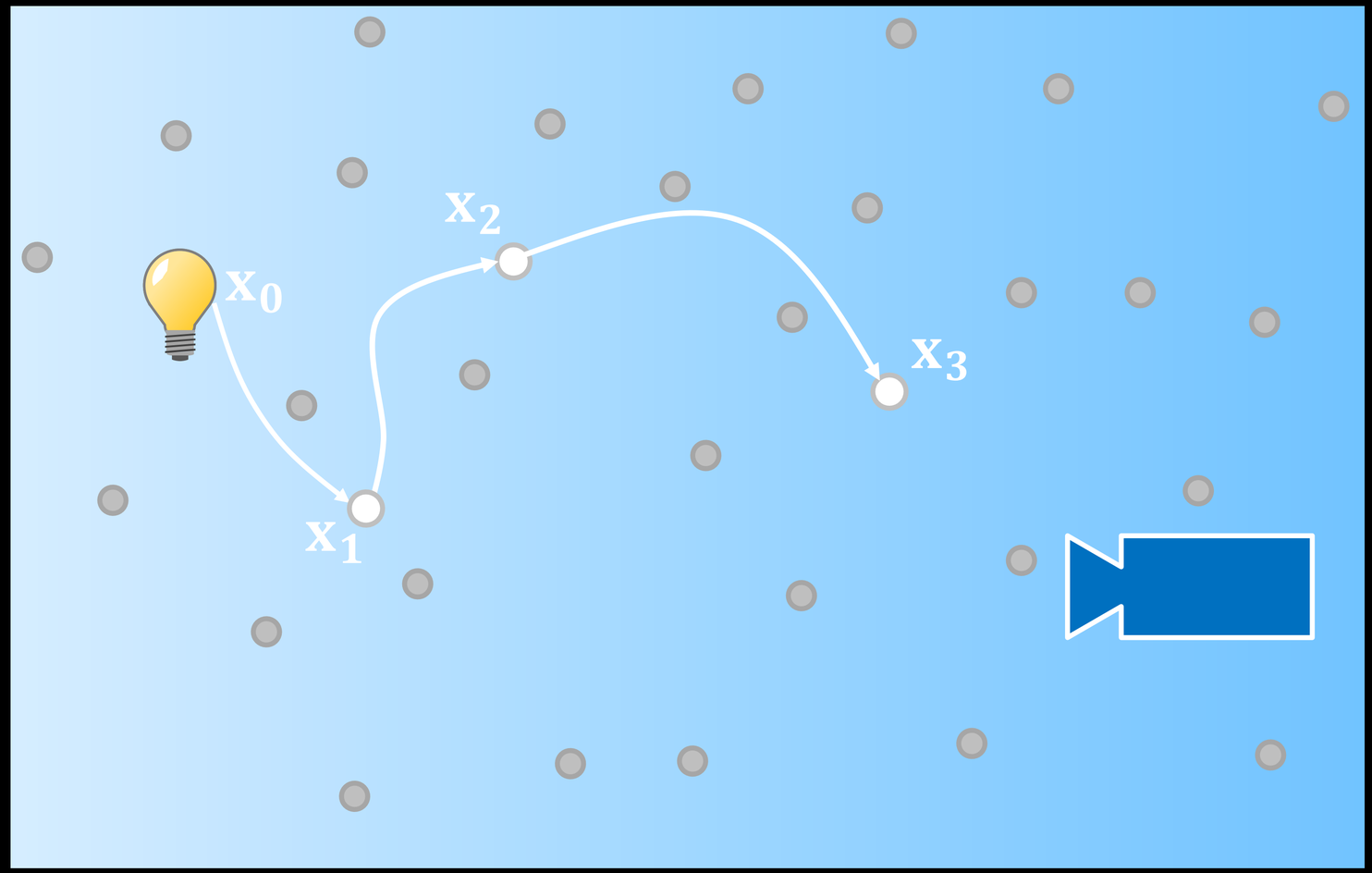
refractive ray tracing

[Kravtsov and Orlov, Book, 1990]
[Gröller, Visual Comp., 1995]
[Stam and Languénou, Rend. Techn., 1996]
[Weiskopf et al., Com. Graph. forum, 2004]
[Guttirez et al., In Rend. Techn., 2005]
[Ihrke et al., ToG, 2007]
[Atcheson et al., ToG, 2008]
[Ji et al., CVPR, 2013]
[Pedrotti et al., Book, 2017]
[Scopelliti et al., Nature LSA, 2019]



Rendering continuous refraction and scattering

radiative transfer equation

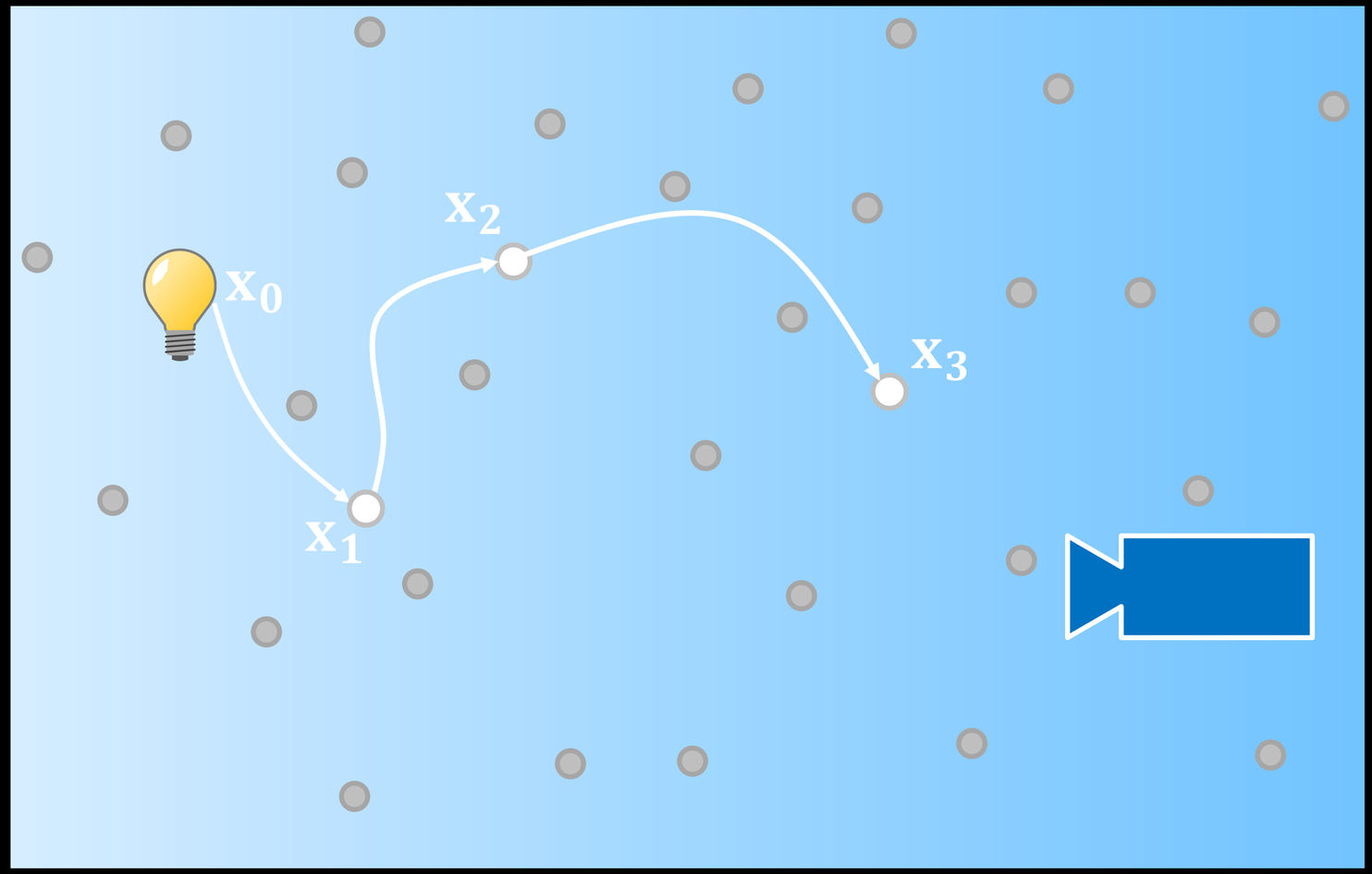


[Chandrasekhar, book, 1960]
[Lenoble, book, 1985]
[Lafortune and Willems, 1996]
[Cammarano and Jensen, 2002]
[Guttirez et al., Com. and Graph. 2006]
[Jarosz et al., Comp. Graph. forum, 2008]
[Jakob et al. ,ToG, 2010]
[Jarosz et al., ToG, 2011]
[Pediredla et al., JBO, 2016]
[Novak et al., Comp. Graph. forum, 2018]
[Bitterli et al., ToG, 2018]

Rendering continuous refraction and scattering

radiative transfer equation

bidirectional path tracing (BDPT):

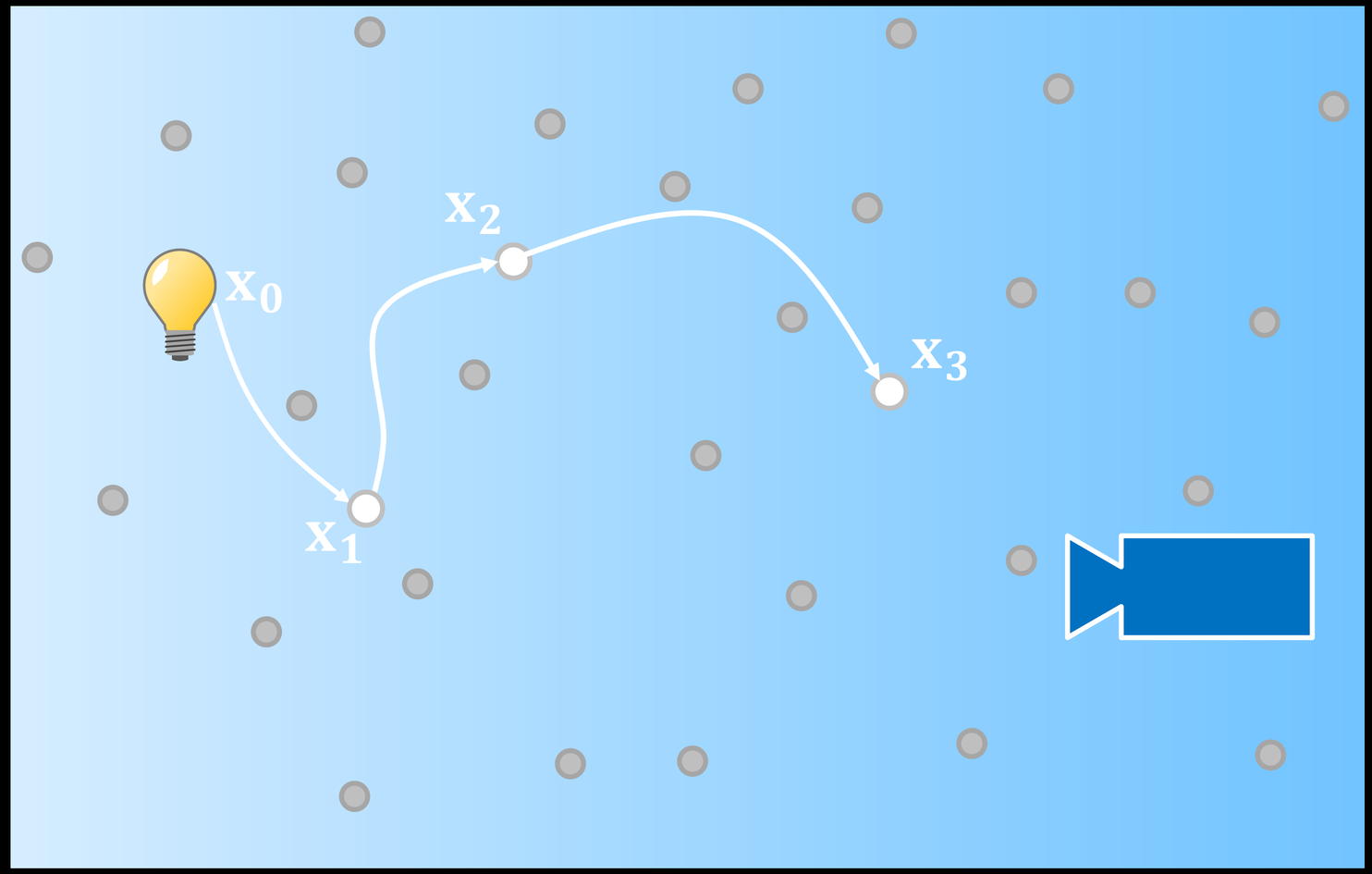


Rendering continuous refraction and scattering

radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath



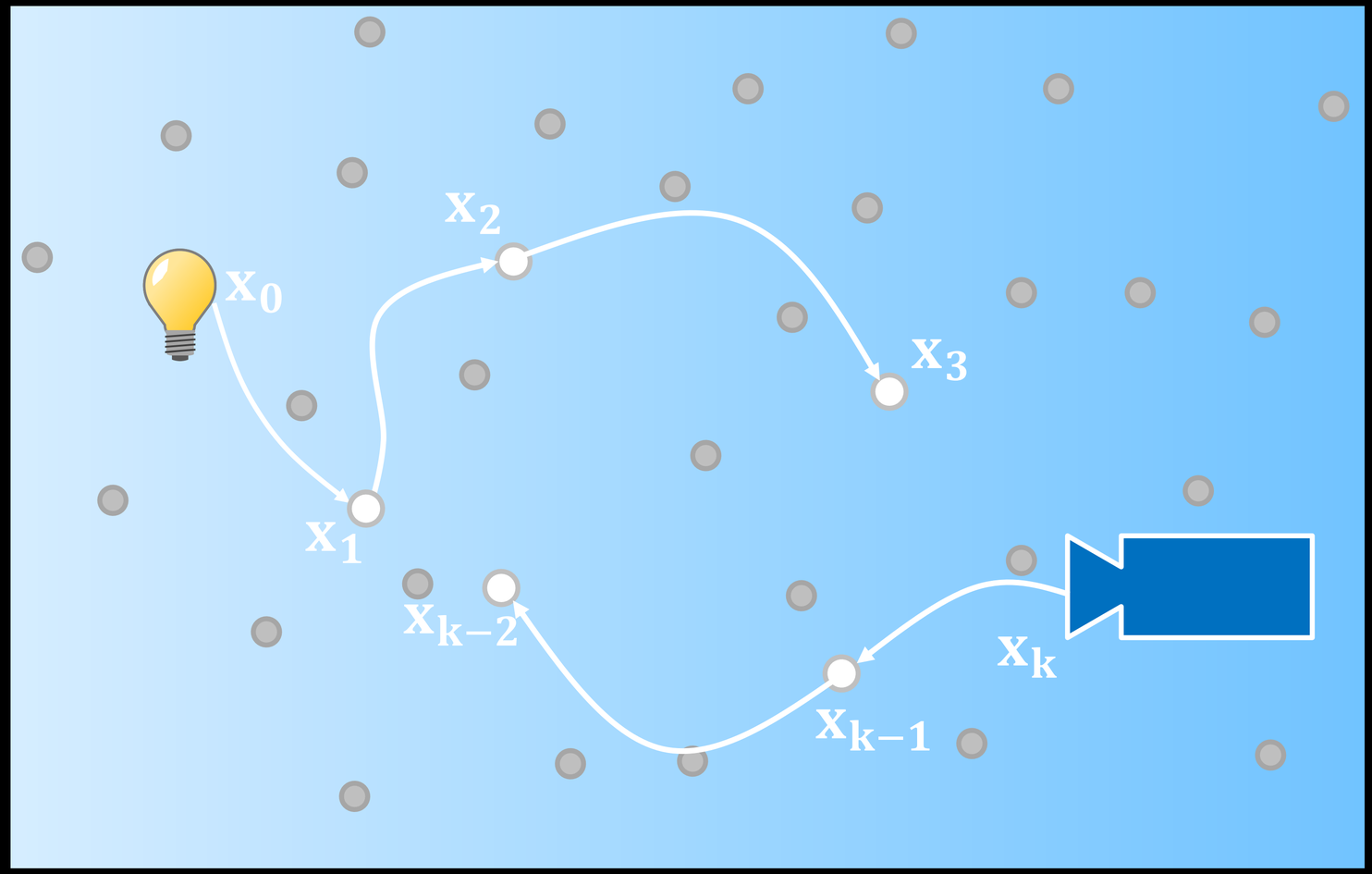
Rendering continuous refraction and scattering

radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath

2. trace a random sensor subpath



Rendering continuous refraction and scattering

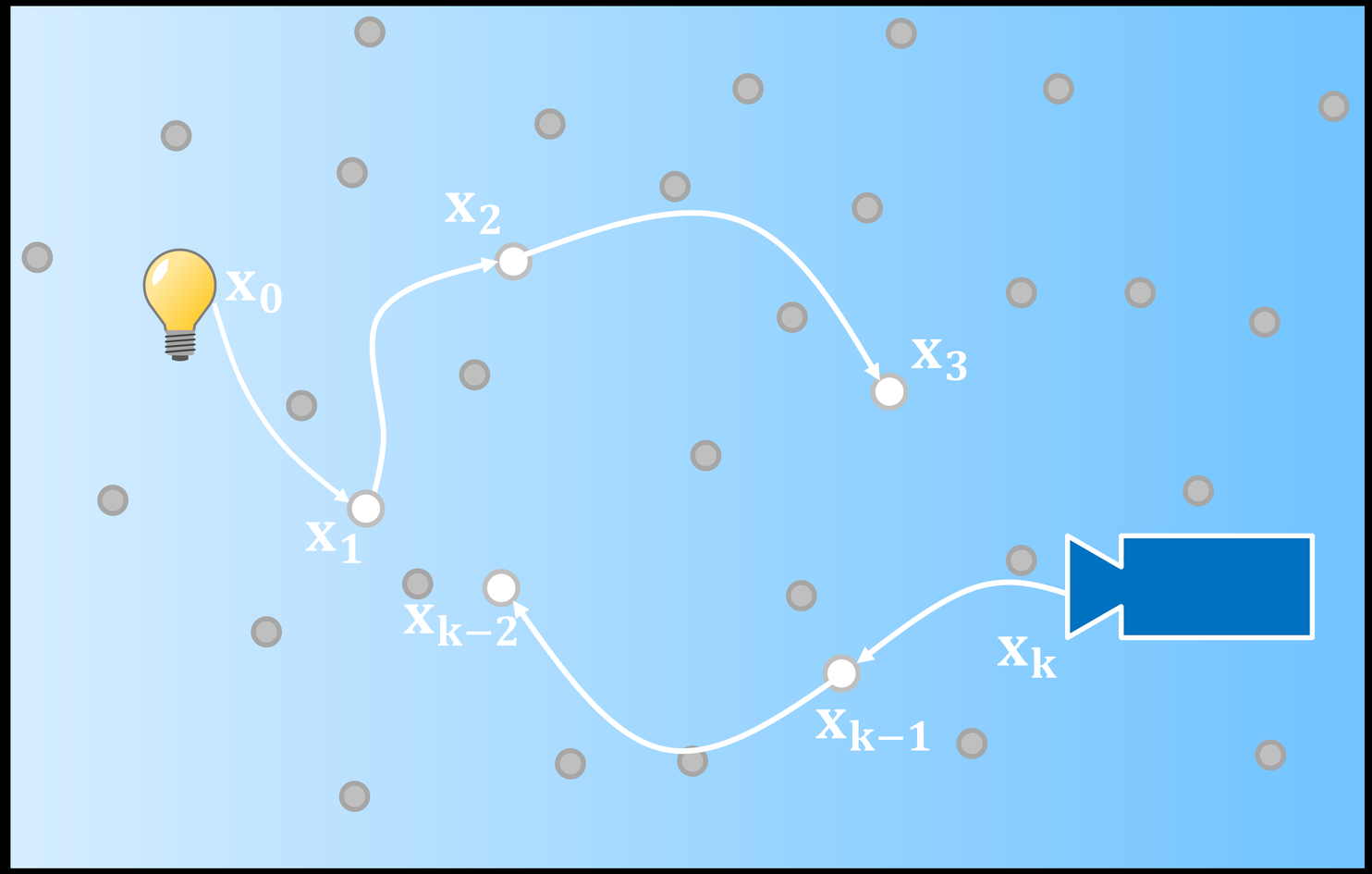
radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath

2. trace a random sensor subpath

3. join vertices with a straight line



Rendering continuous refraction and scattering

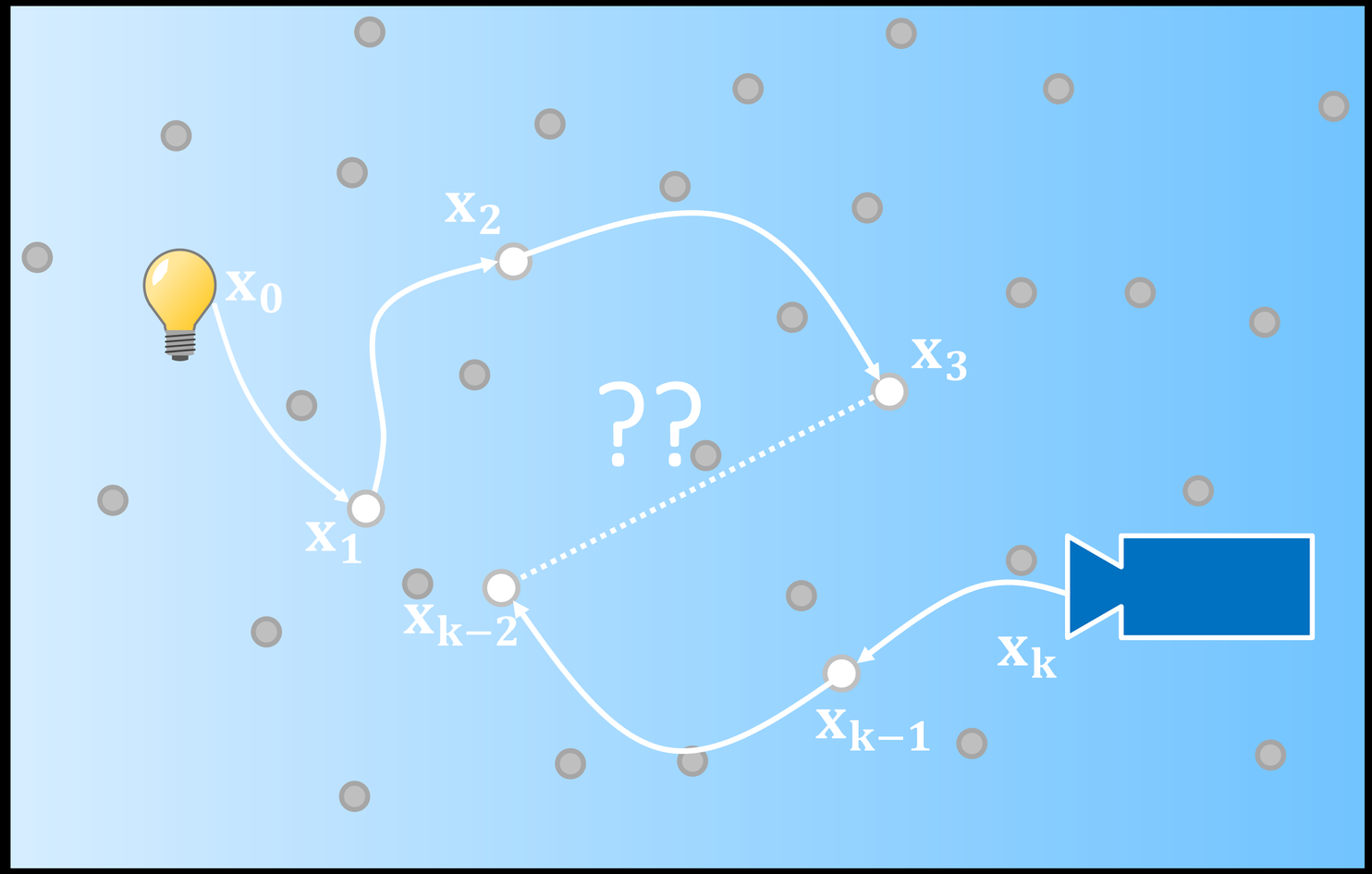
radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath

2. trace a random sensor subpath

3. join vertices with a ~~straight line~~ curve



Rendering continuous refraction and scattering

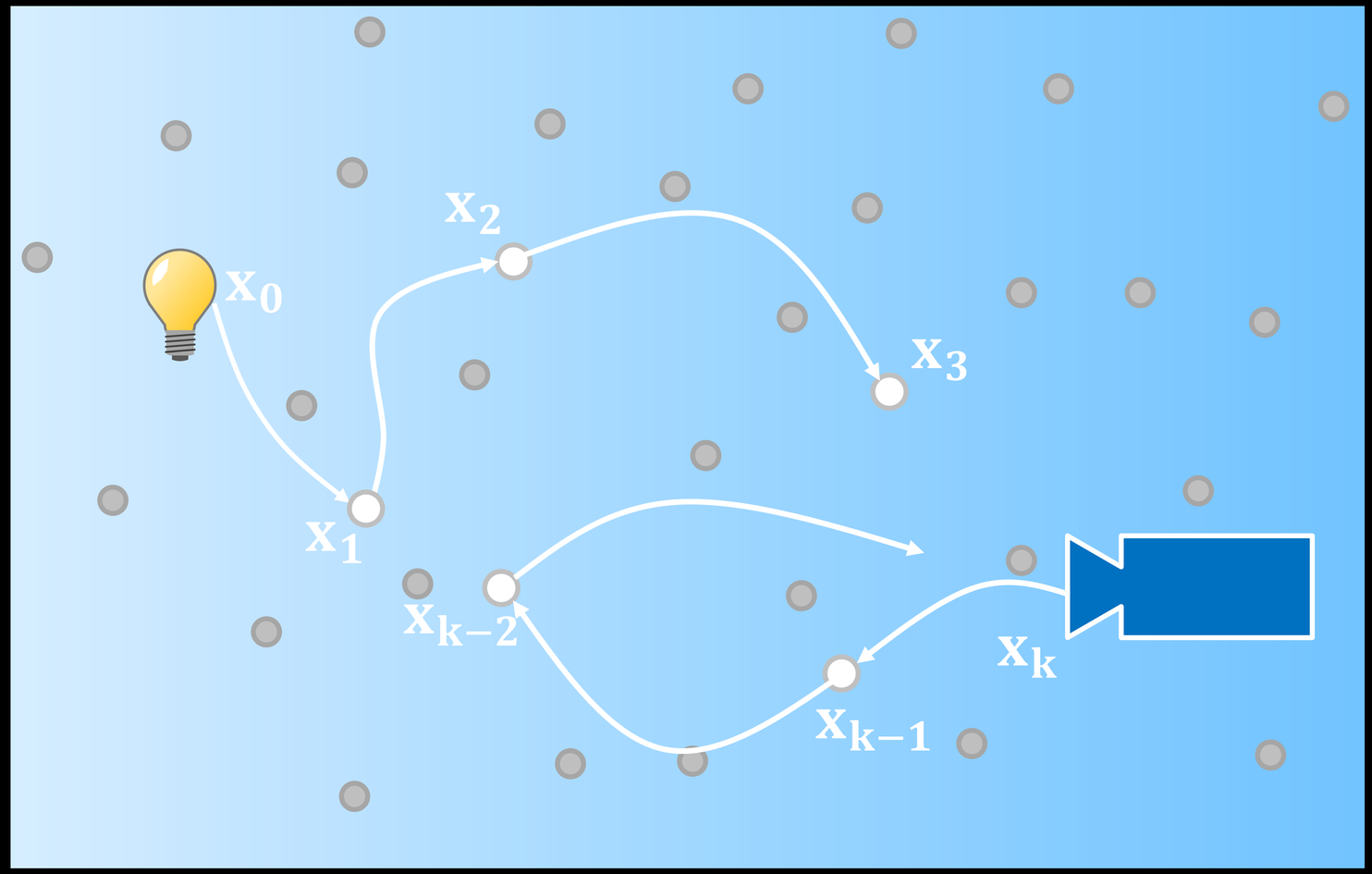
radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath

2. trace a random sensor subpath

3. join vertices with a ~~straight line~~ curve



Rendering continuous refraction and scattering

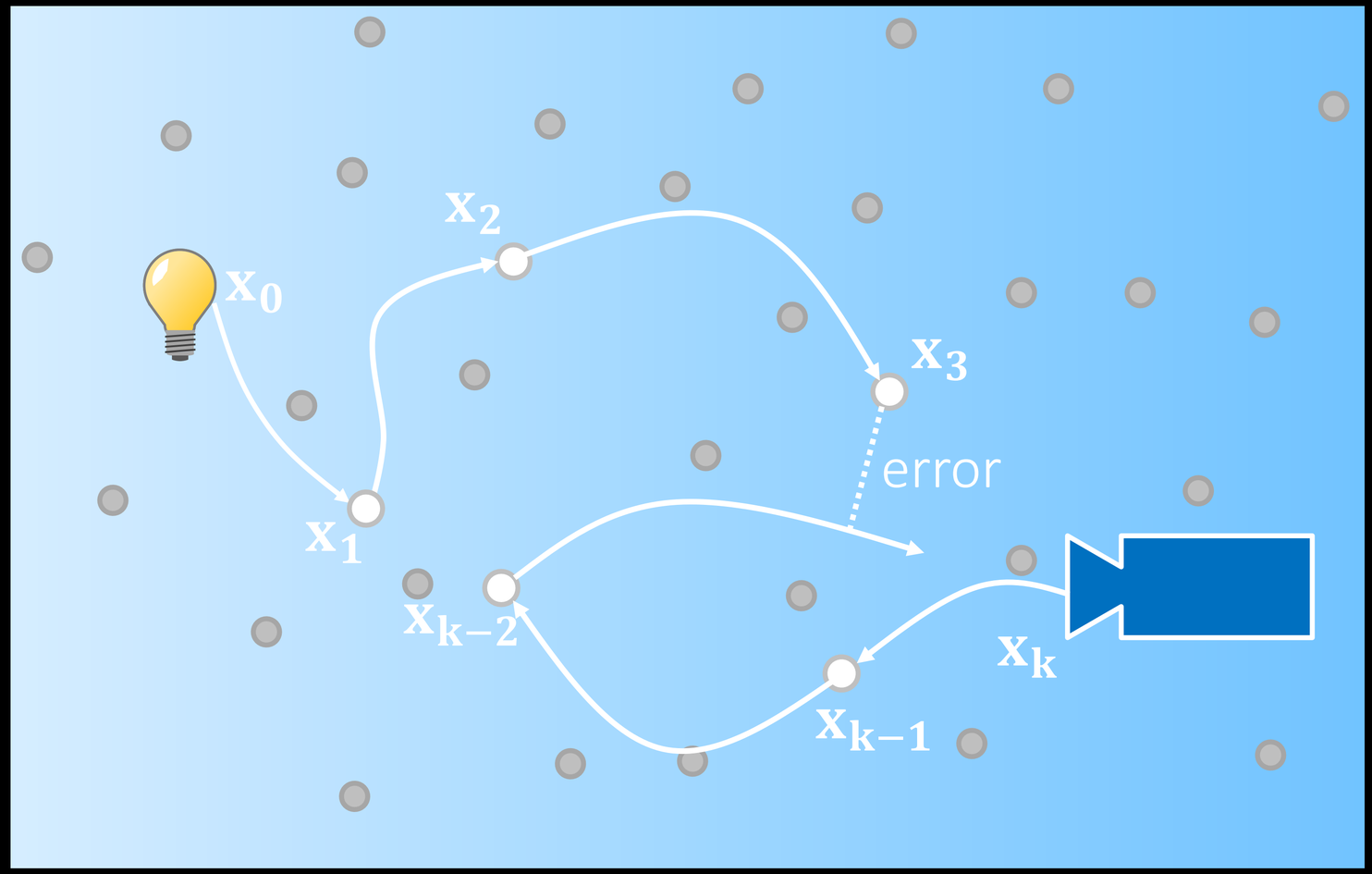
radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath

2. trace a random sensor subpath

3. join vertices with a ~~straight line~~ curve



Rendering continuous refraction and scattering

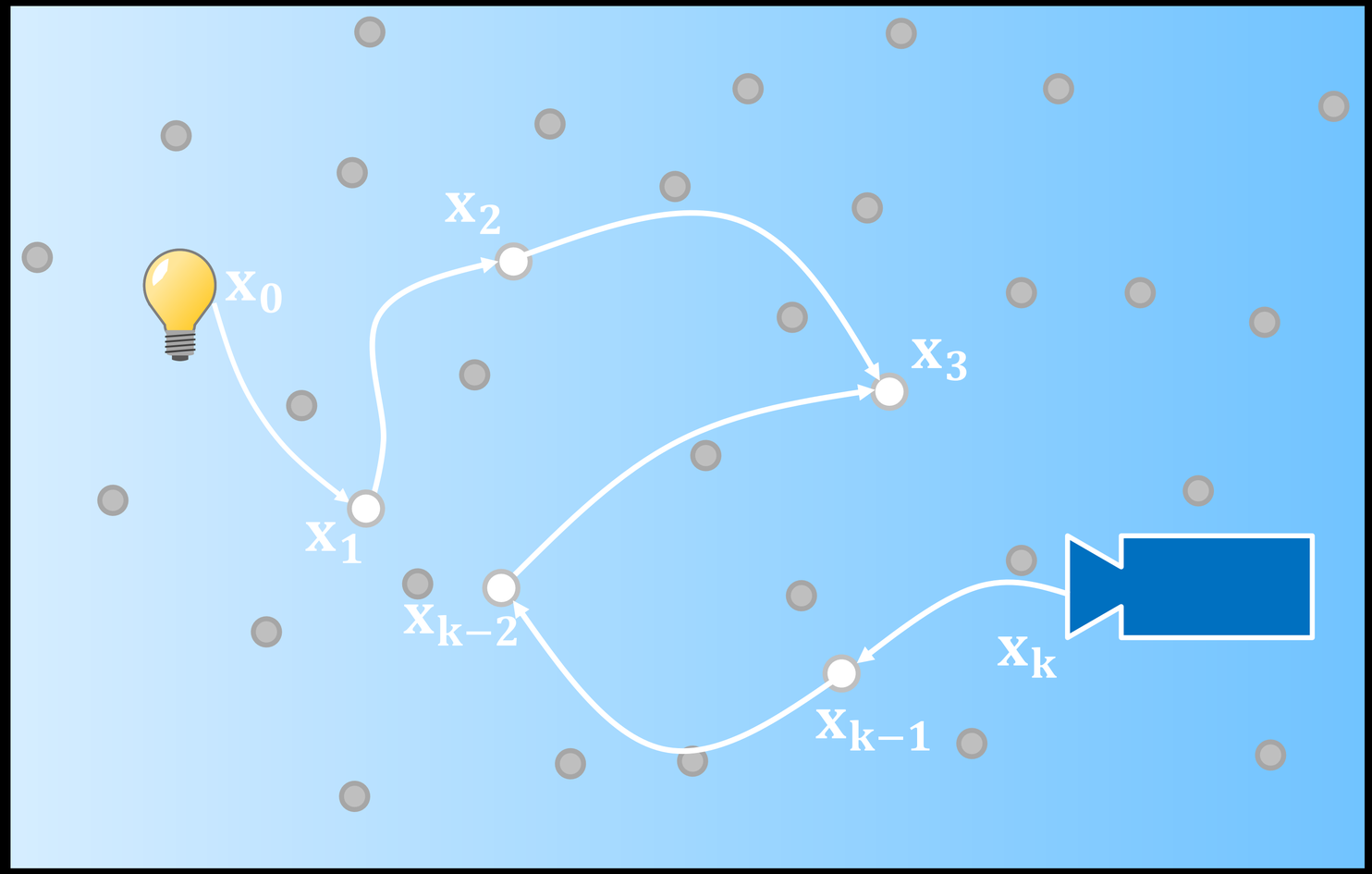
radiative transfer equation

bidirectional path tracing (BDPT):

1. trace a random emitter subpath

2. trace a random sensor subpath

3. join vertices with a ~~straight line~~ curve



Rendering continuous refraction and scattering

$$\frac{d\mathcal{L}}{d\mathbf{v_0}} = (\mathbf{x}_{s\star} - \mathbf{y})^T \frac{d\mathbf{x}_{s\star}}{d\mathbf{v_0}},$$

$$\frac{d\mathbf{x}_{s\star}}{d\mathbf{v_0}} = \frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}} + \frac{\partial \mathbf{x}_{s\star}}{\partial s^\star}\frac{ds^\star}{d\mathbf{v_0}}.$$

$$g(\mathbf{x}_{s\star}) = 0 \Rightarrow \frac{dg(\mathbf{x}_{s\star})}{d\mathbf{x}_{s\star}}\frac{d\mathbf{x}_{s\star}}{d\mathbf{v_0}} = 0$$

$$\Rightarrow \frac{ds^\star}{d\mathbf{v_0}} = -\frac{\frac{dg(\mathbf{x}_{s\star})}{d\mathbf{x}_{s\star}}\frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}}}{\frac{dg(\mathbf{x}_{s\star})}{d\mathbf{x}_{s\star}}\frac{\partial \mathbf{x}_{s\star}}{\partial s^\star}},$$

$$\frac{d\mathbf{x}_{s\star}}{d\mathbf{v_0}} = \left(I_{3\times 3} - \frac{\frac{\partial \mathbf{x}_{s\star}}{\partial s^\star}\frac{dg(\mathbf{x}_{s\star})}{d\mathbf{x}_{s\star}}}{\frac{dg(\mathbf{x}_{s\star})}{d\mathbf{x}_{s\star}}\frac{\partial \mathbf{x}_{s\star}}{\partial s^\star}}\right)\frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}}.$$

Algorithm 2: Symplectic integration for derivative tracing

Input: $n(\mathbf{x}), \nabla n(\mathbf{x}), H_n(\mathbf{x}), \text{ray}(\mathbf{x}, \mathbf{v}), \frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}} = O_3, \frac{\partial \mathbf{v}_{s\star}}{\partial \mathbf{v_0}} = I_3,$ nSteps, s = step size

Output: $\frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}}, \frac{\partial \mathbf{v}_{s\star}}{\partial \mathbf{v_0}}$

for $i = 1 : nSteps$ **do**

$\text{ray.v} \mathrel{+}= 0.5s\nabla n(\text{ray.x});$

$\frac{\partial \mathbf{v}_{s\star}}{\partial \mathbf{v_0}} \mathrel{+}= 0.5sH_n(\text{ray.x})\frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}};$

$\text{ray.x} \mathrel{+}= s\frac{\text{ray.v}}{n(\text{ray.x})};$

$\frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}} \mathrel{+}= s\left(-\frac{\mathbf{v_s}\nabla n(\text{ray.x})}{n(\text{ray.x})^2}\frac{\partial \mathbf{x_s}}{\partial \mathbf{v_0}} + \frac{1}{n(\text{ray.x})}\frac{\partial \mathbf{v_s}}{\partial \mathbf{v_0}}\right);$

$\text{ray.v} \mathrel{+}= 0.5s\nabla n(\text{ray.x});$

$\frac{\partial \mathbf{v}_{s\star}}{\partial \mathbf{v_0}} \mathrel{+}= 0.5sH_n(\text{ray.x})\frac{\partial \mathbf{x}_{s\star}}{\partial \mathbf{v_0}};$

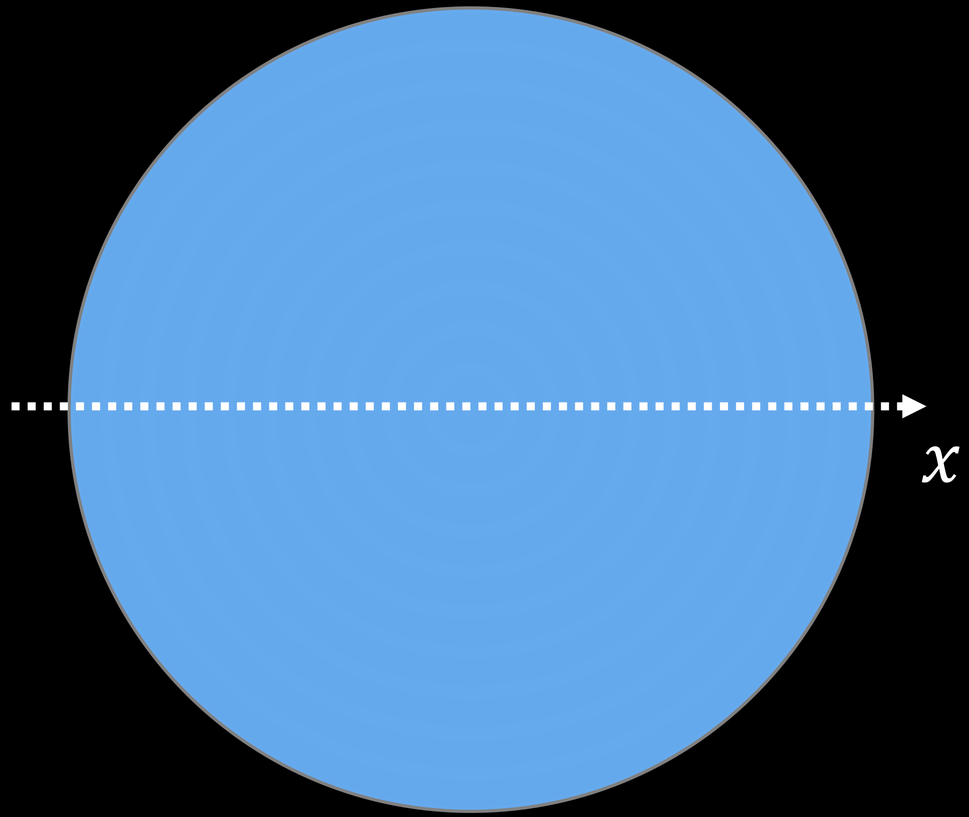
end

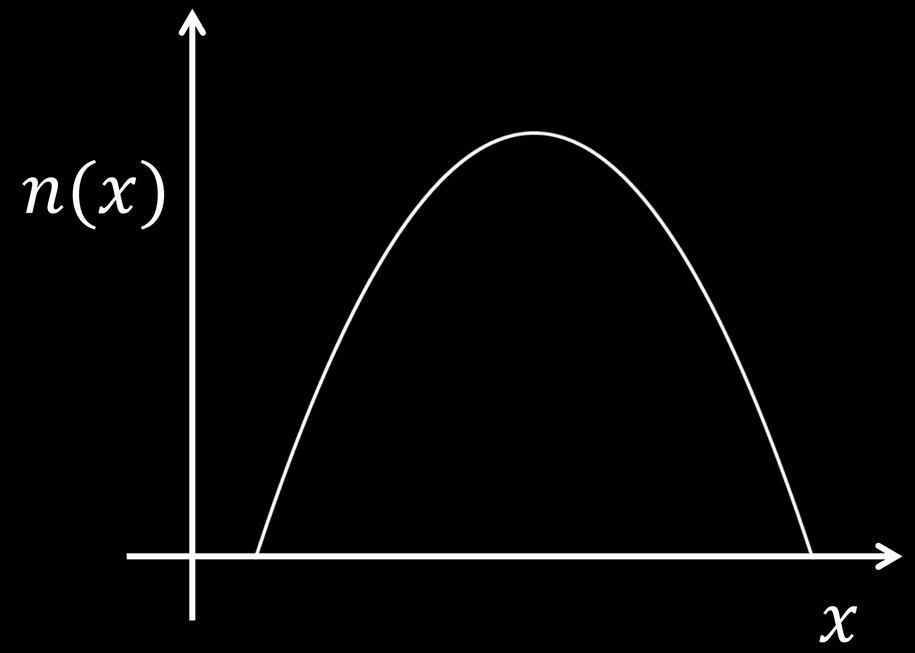
Pediredla et al. *Path Tracing Estimators for Refractive Radiative Transfer,* TOG 2020

Application: simulate Luneburg lenses

https://en.wikipedia.org/wiki/Luneburg_lens

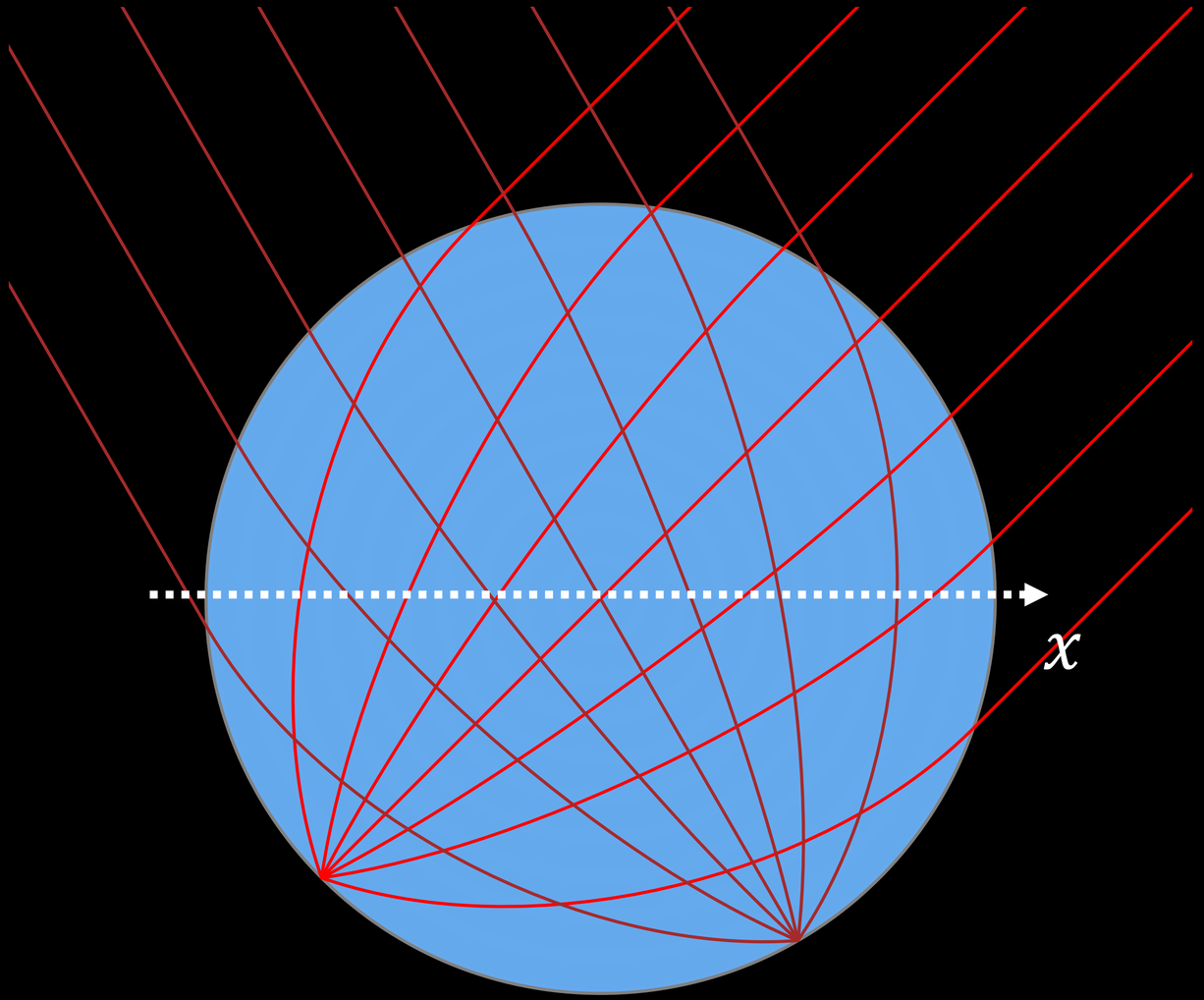
Application: simulate Luneburg lenses

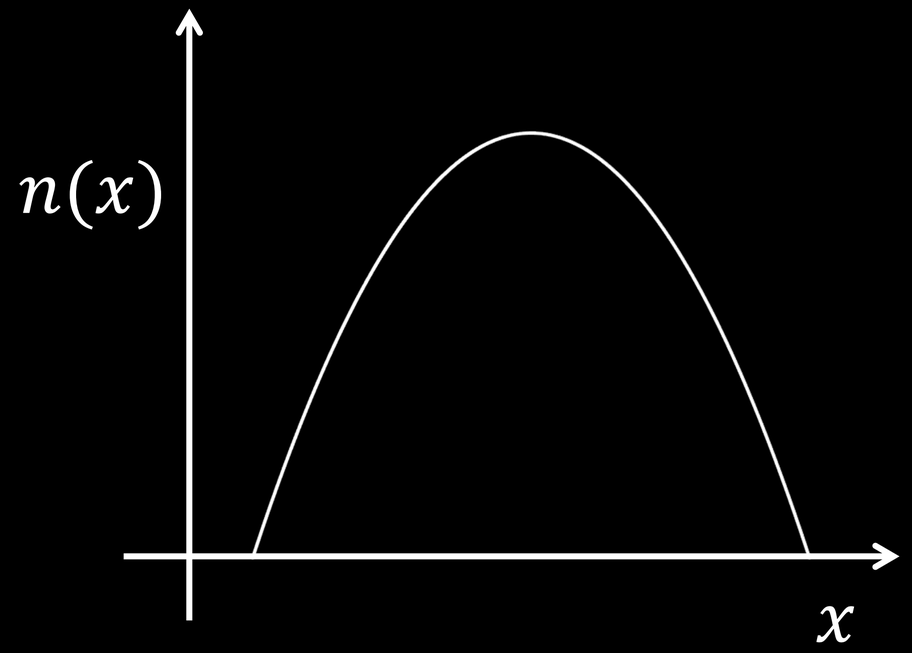




https://en.wikipedia.org/wiki/Luneburg_lens

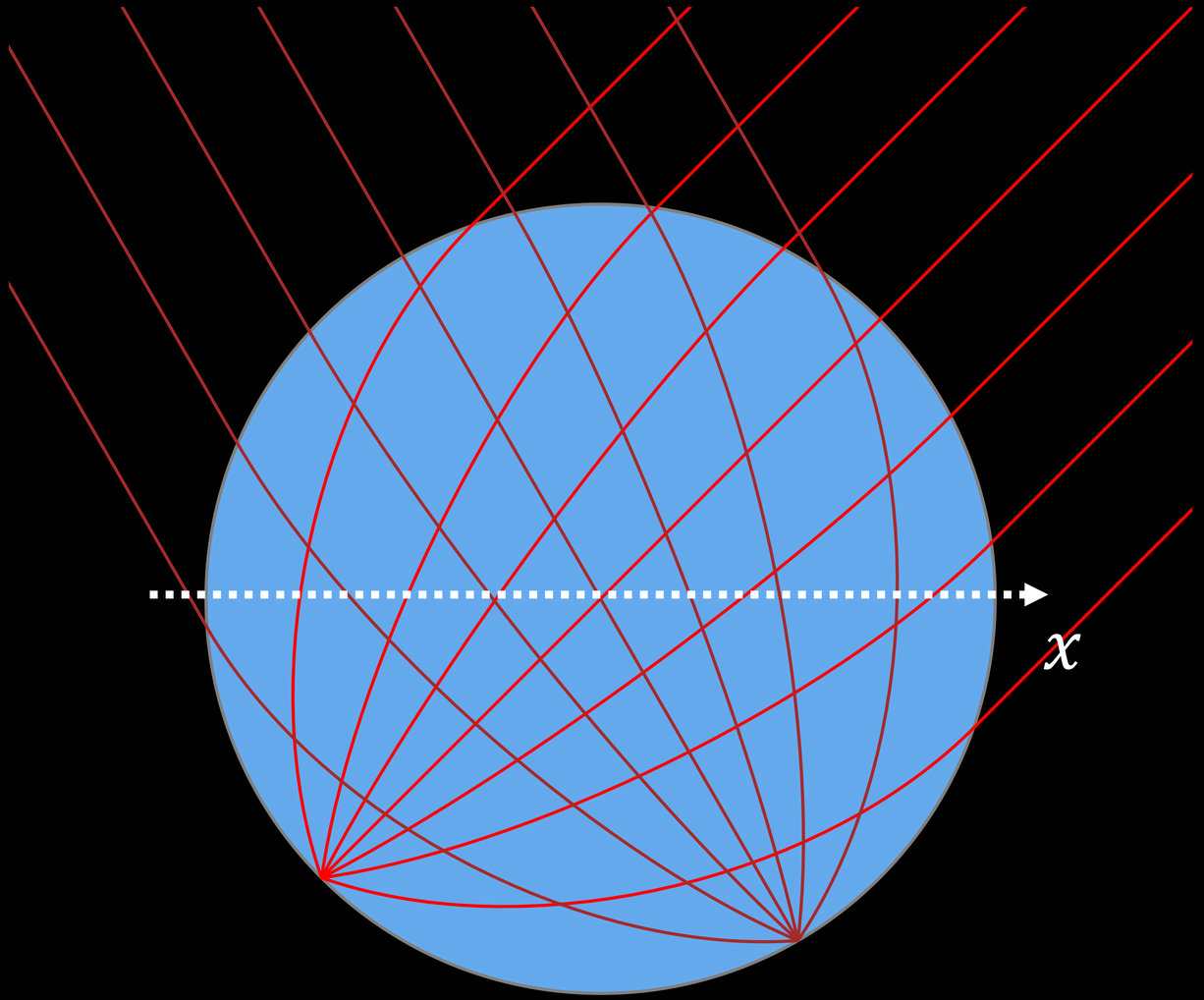
Application: simulate Luneburg lenses

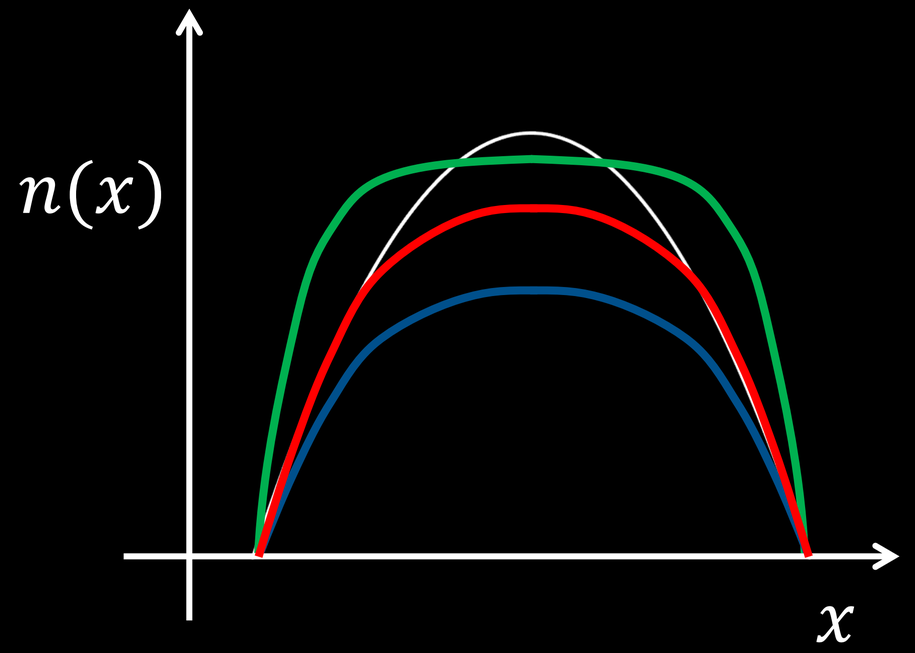




https://en.wikipedia.org/wiki/Luneburg_lens

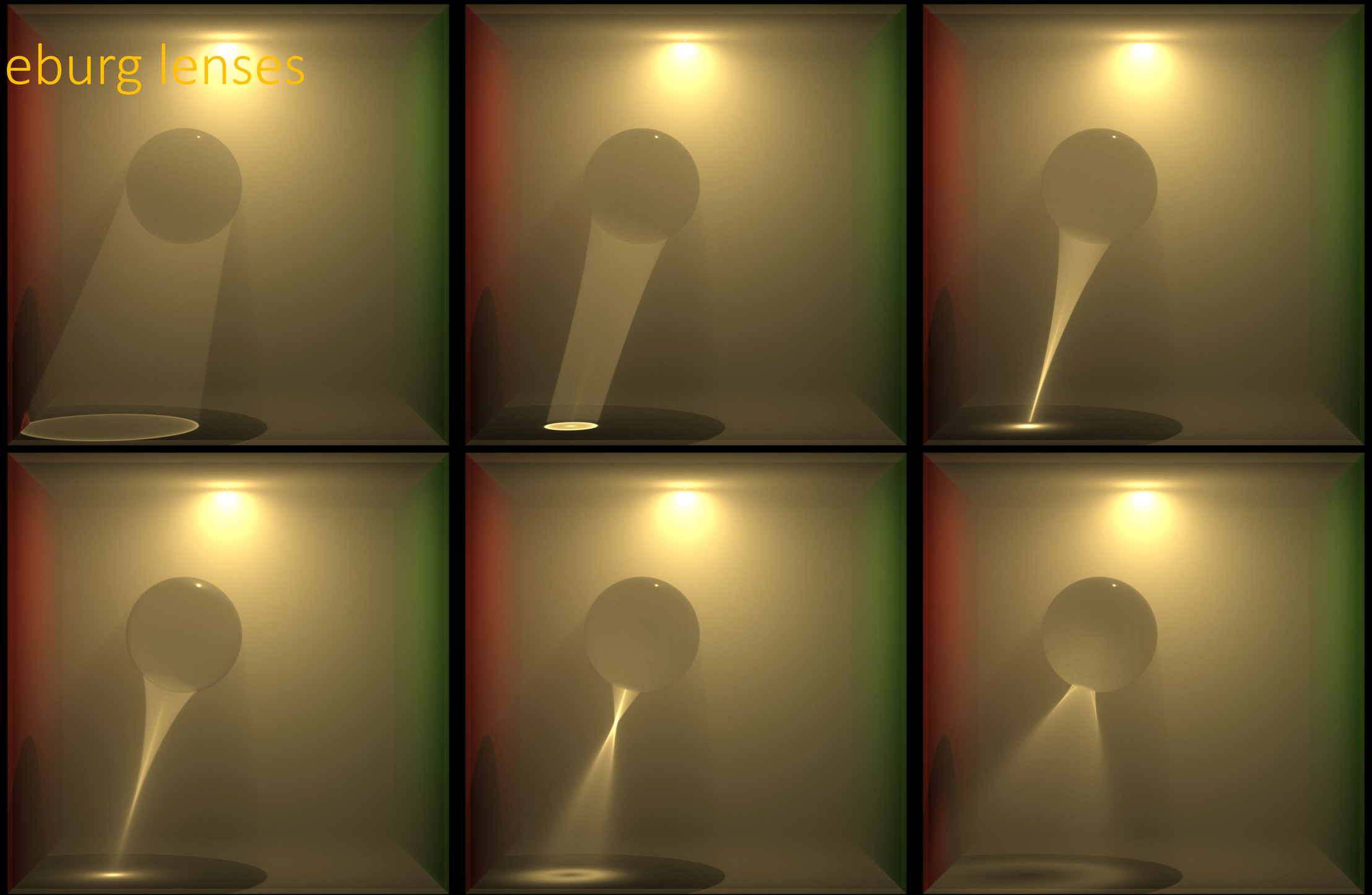
Application: simulate Luneburg lenses





https://en.wikipedia.org/wiki/Luneburg_lens

Luneburg lenses



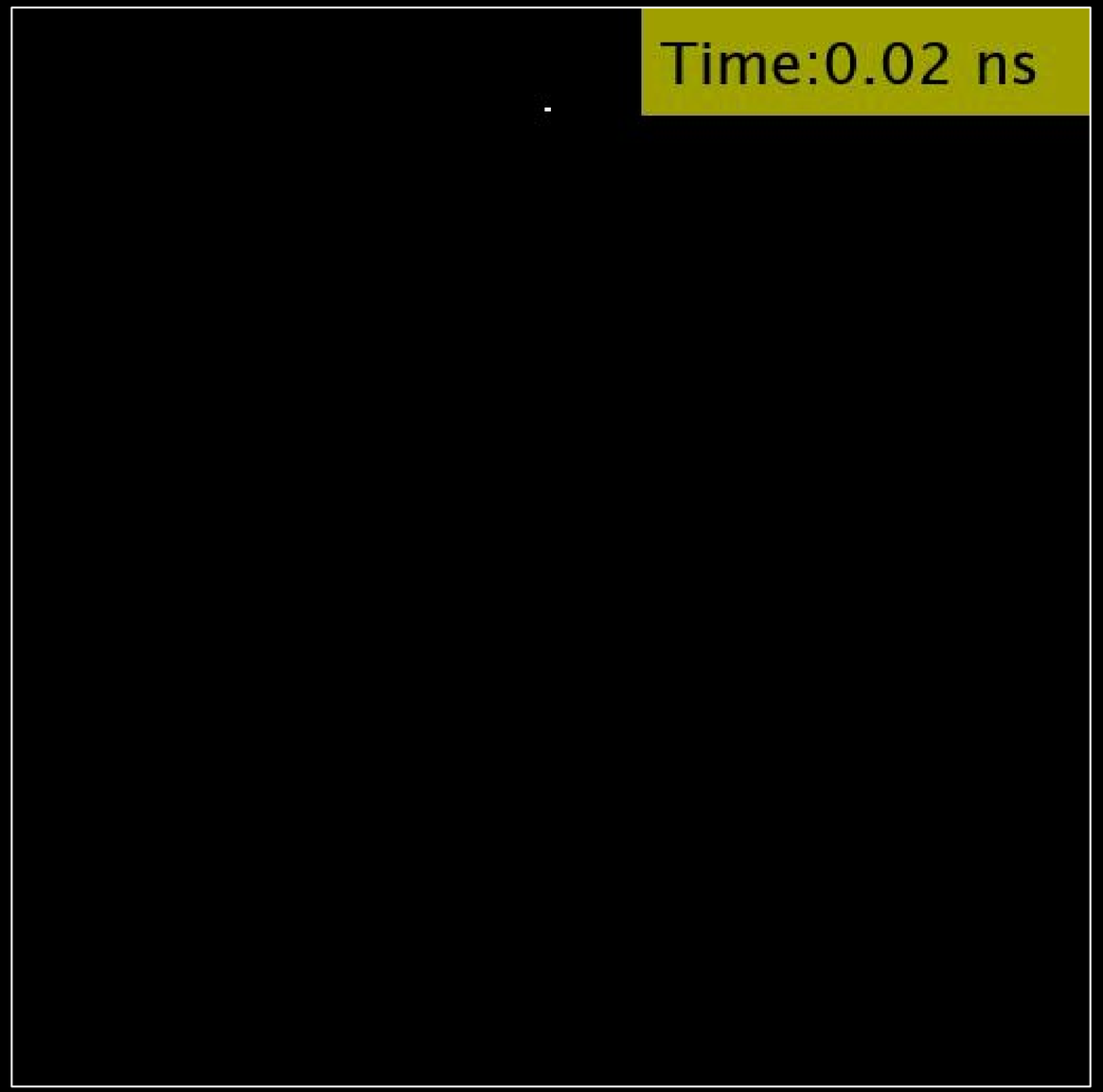
Application: transient rendering

constant refractive index

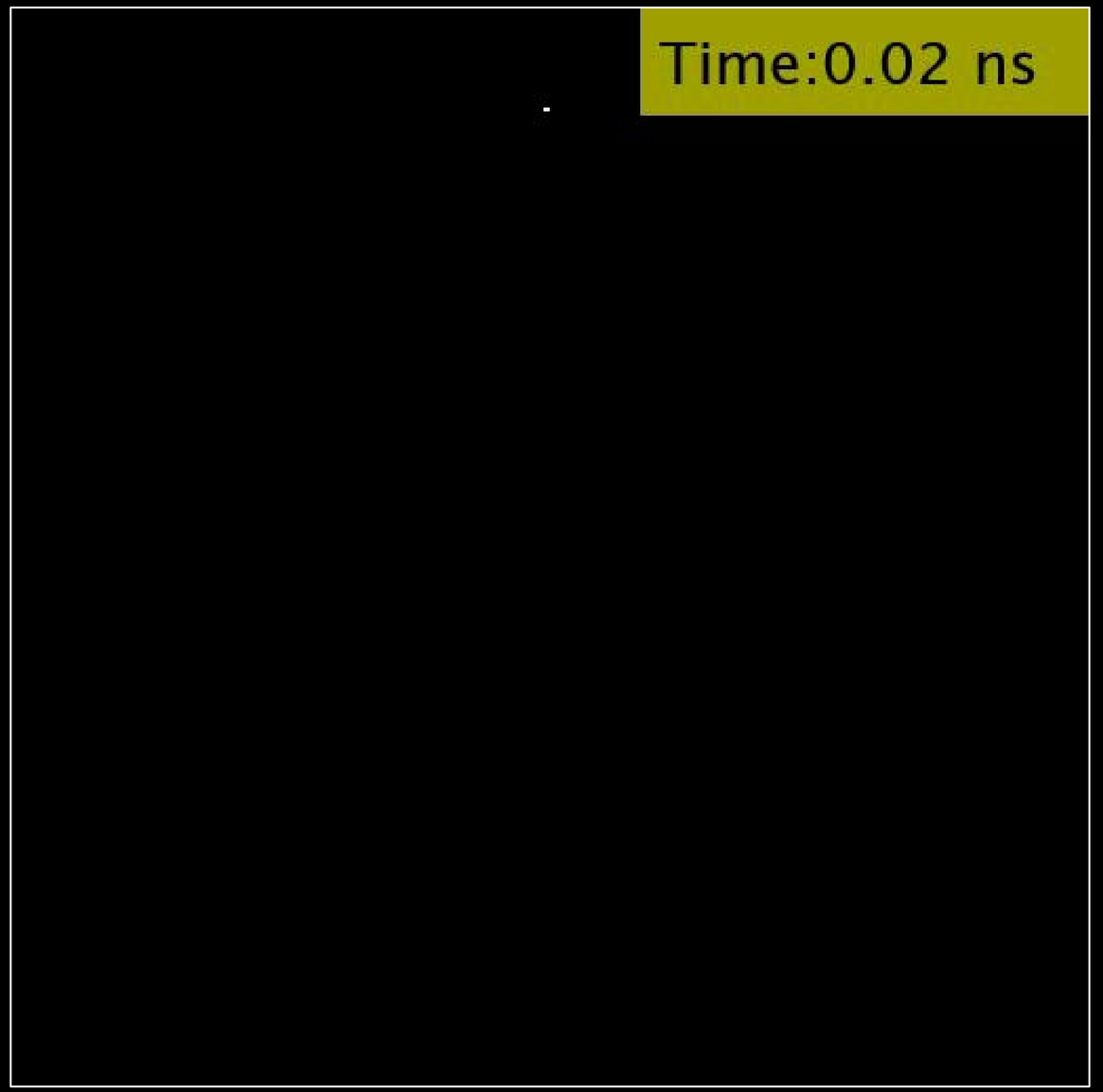
continuous refractive index

Application: transient rendering

constant refractive index

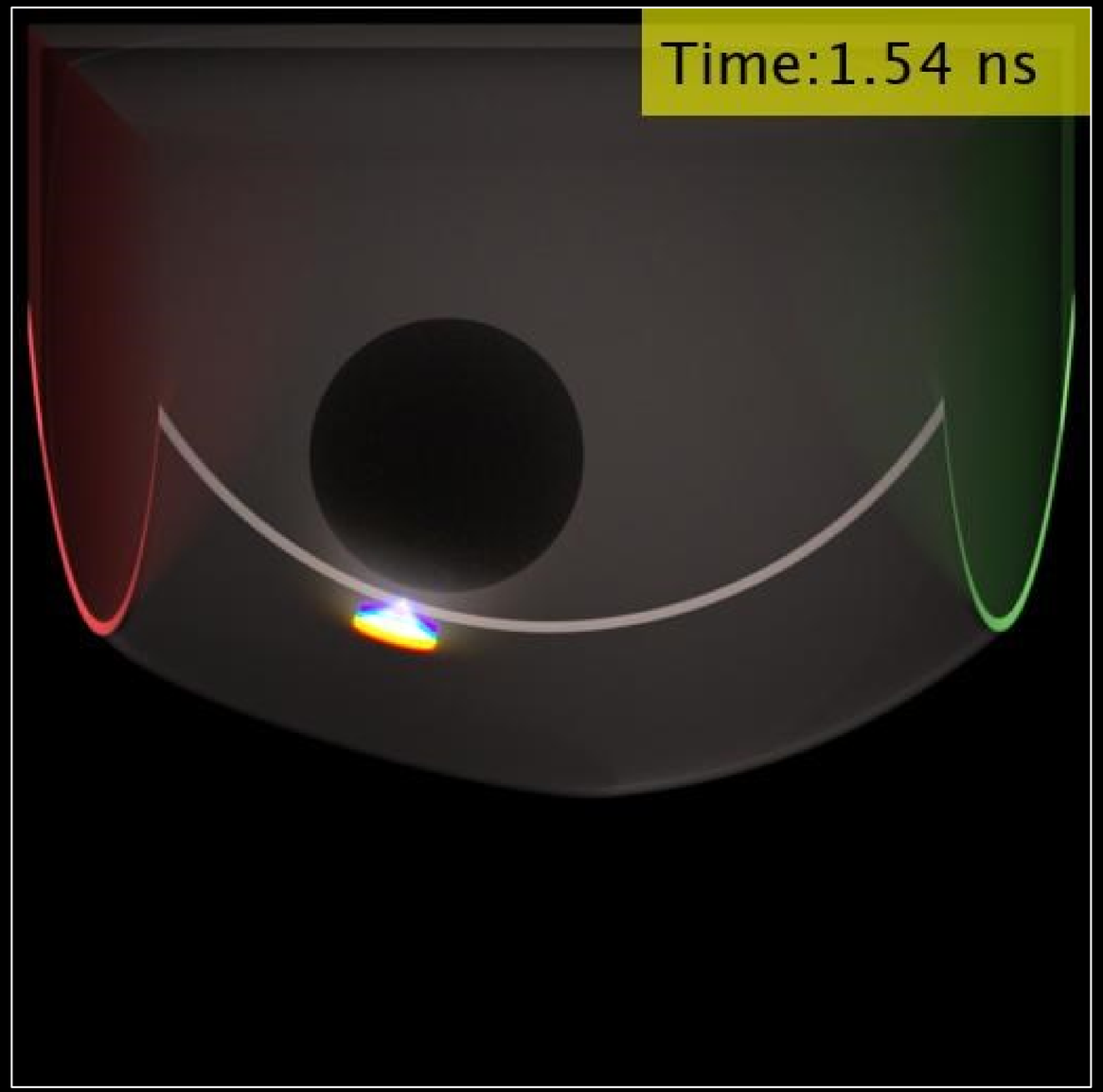


continuous refractive index

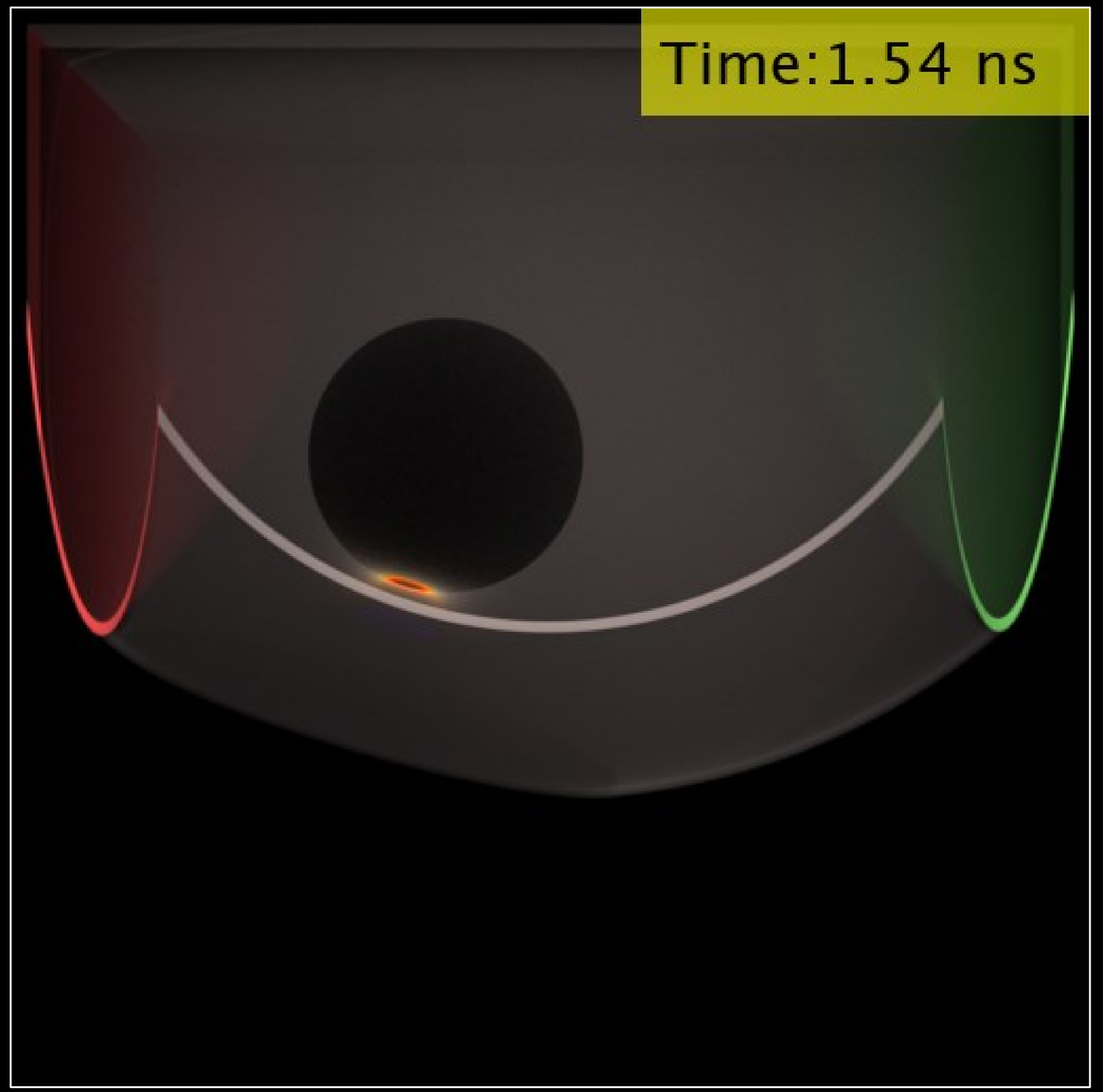


Application: transient rendering

constant refractive index

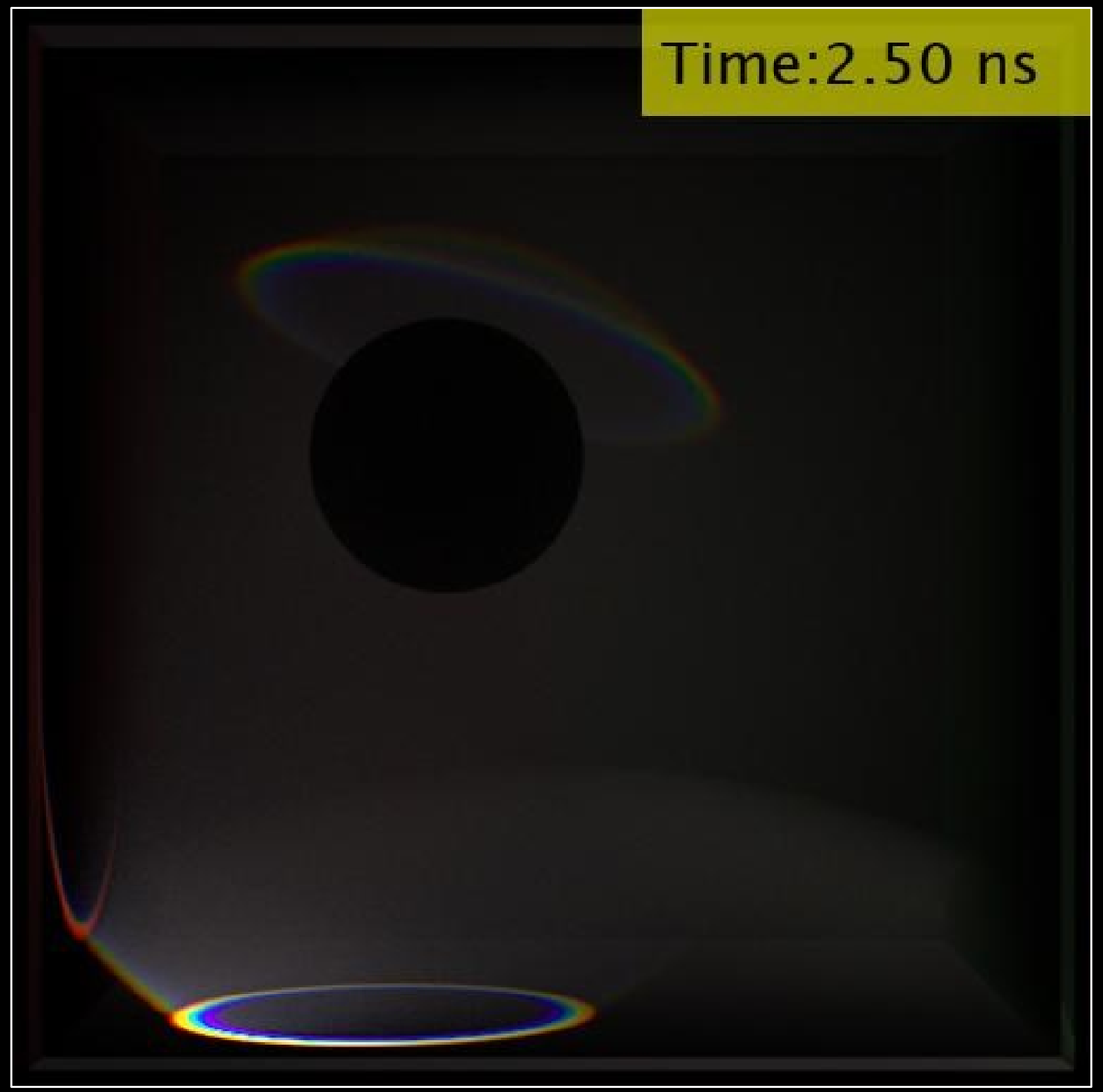


continuous refractive index

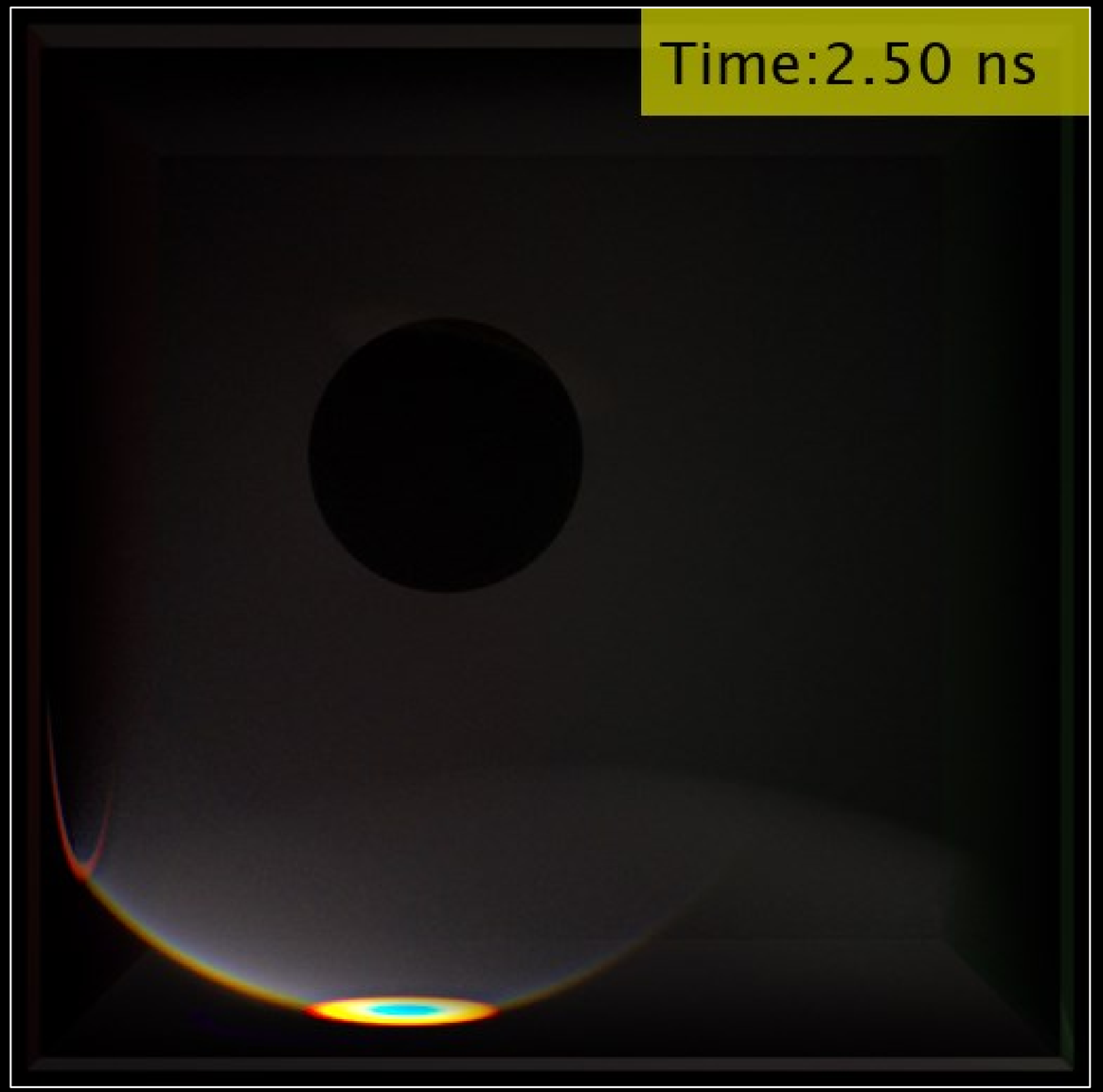


Application: transient rendering

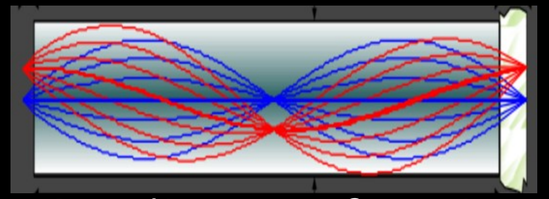
constant refractive index



continuous refractive index



Application: focusing light inside tissue

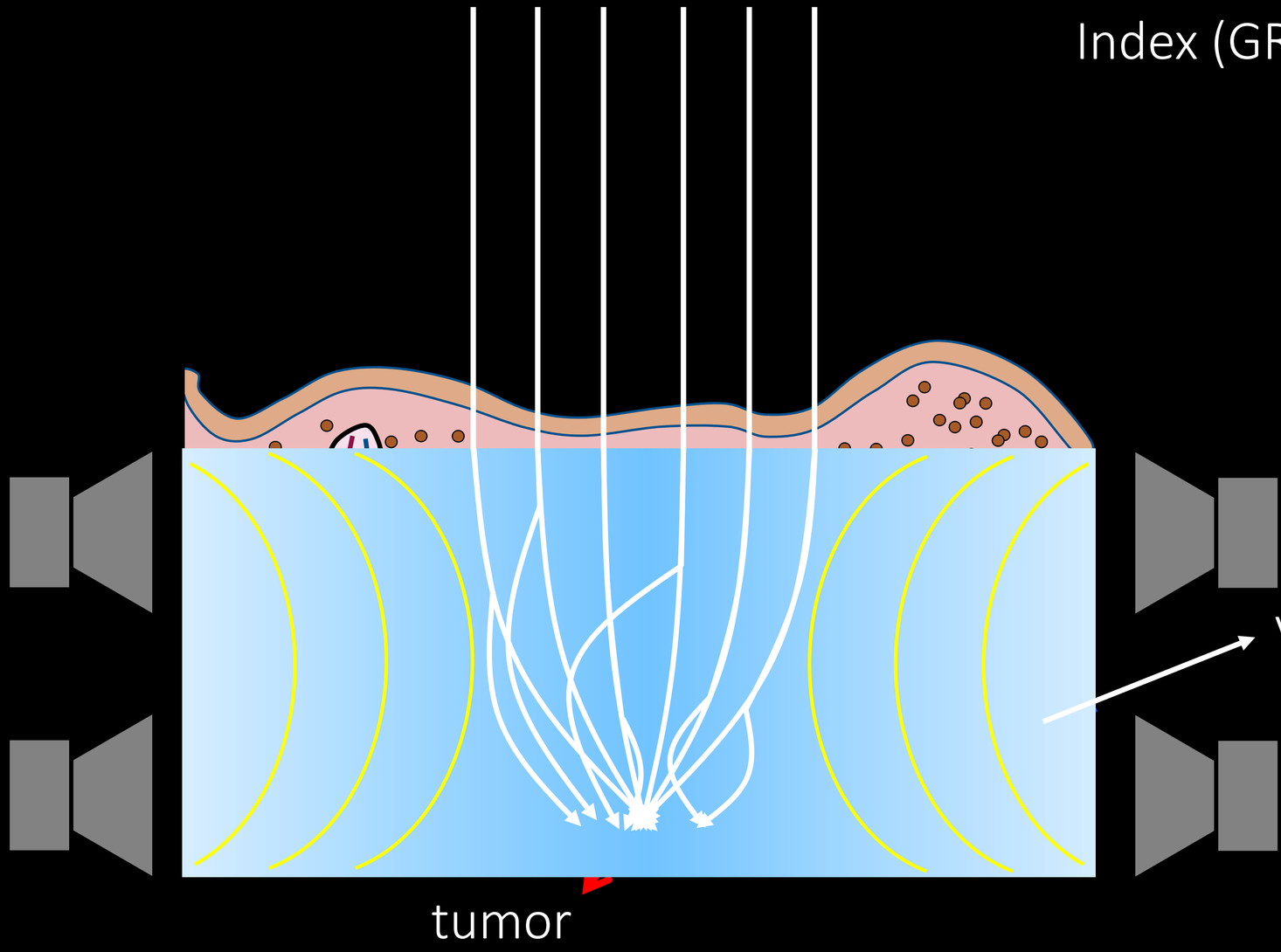


Gradient Refractive Index (GRIN) waveguide

High-dimensional, highly-non-linear design problem:

- ultrasound frequency
- ultrasound voltage
- placement of transducers
- waveform shape
- and more…

Efficiently explore using rendering

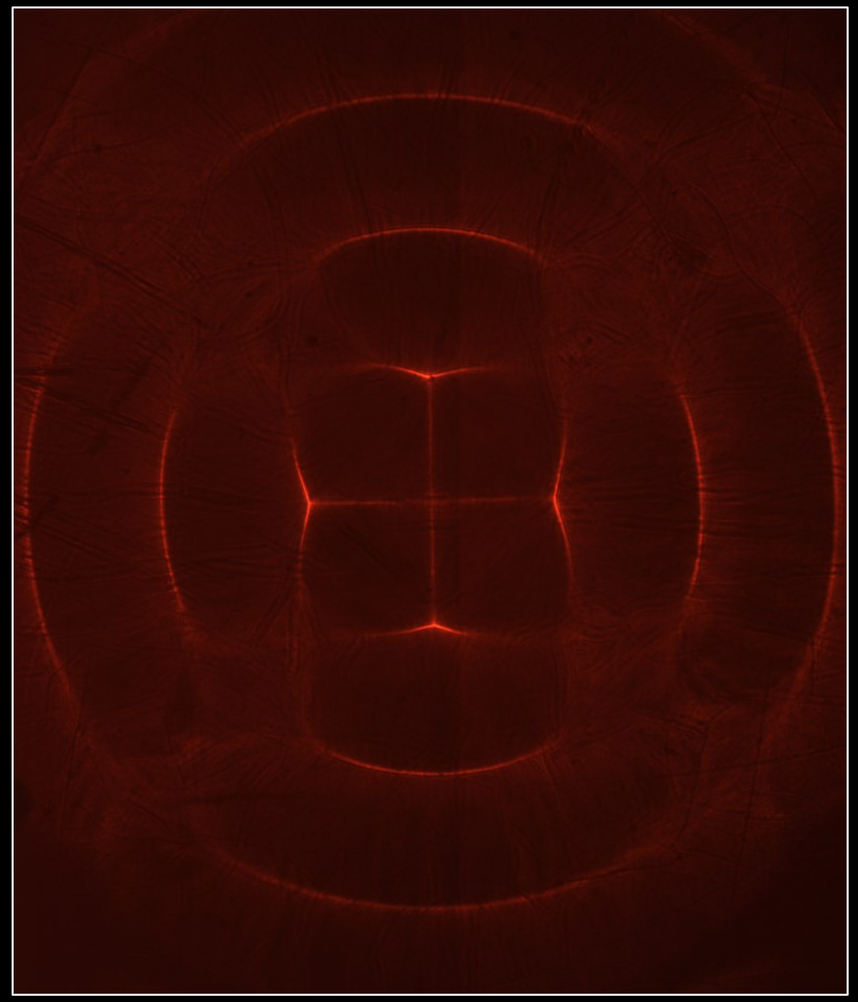


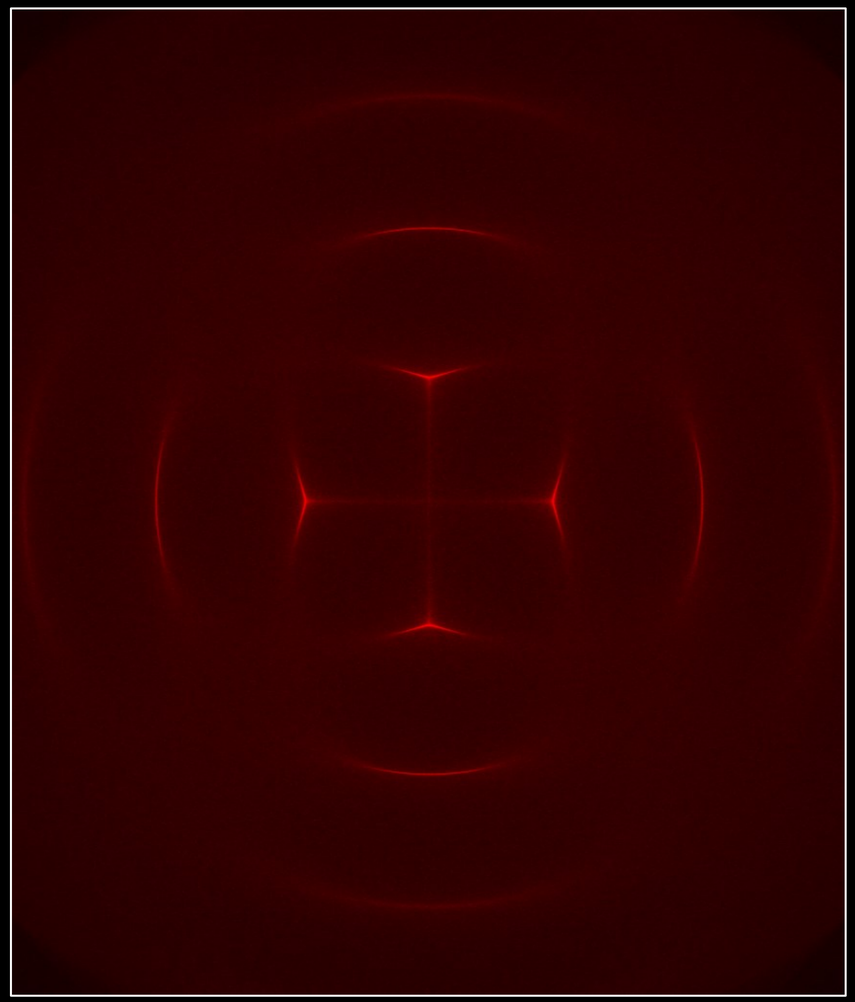
Rendering virtual ultrasonic waveguides

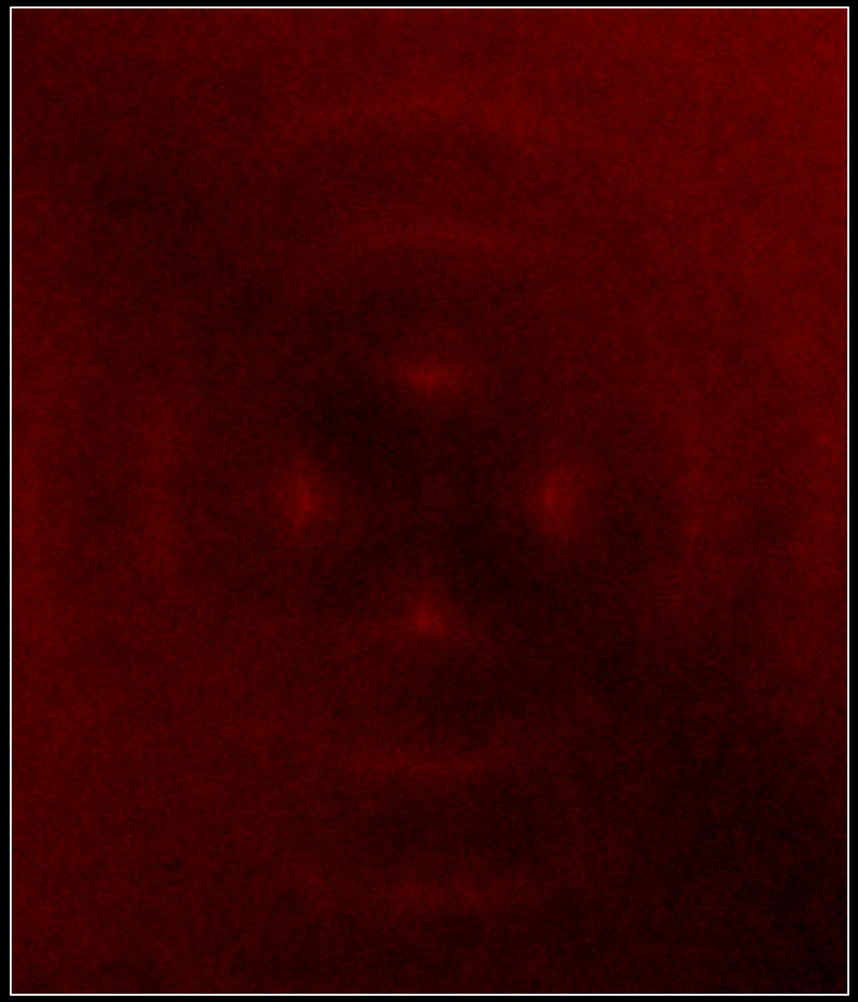
real measurement

BDPT
(our technique)

photon mapping
(previous technique)



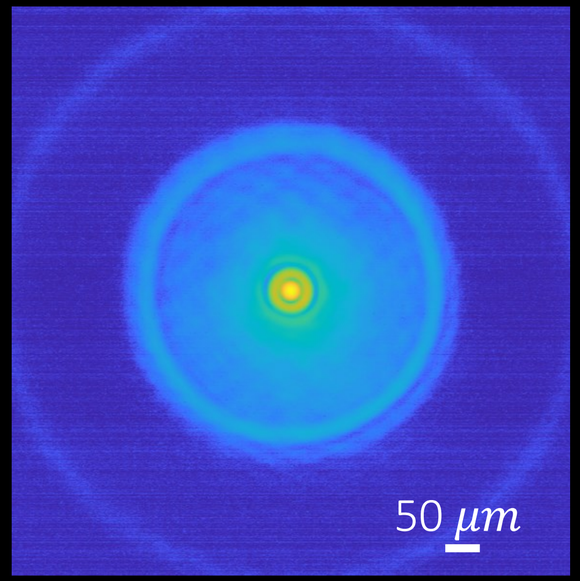




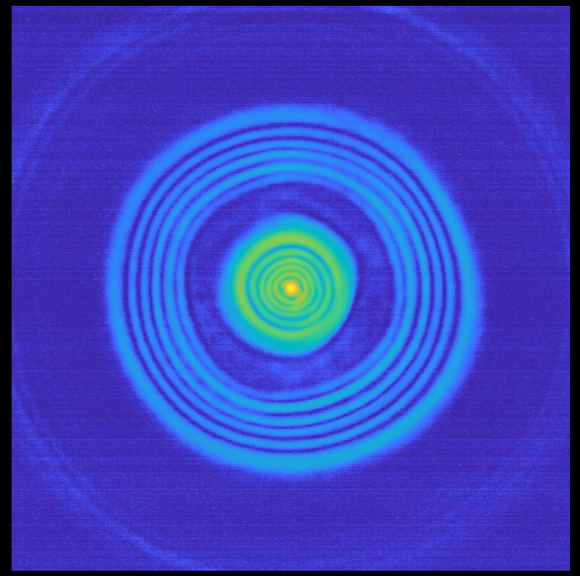
Validation of simulated data

experimental data

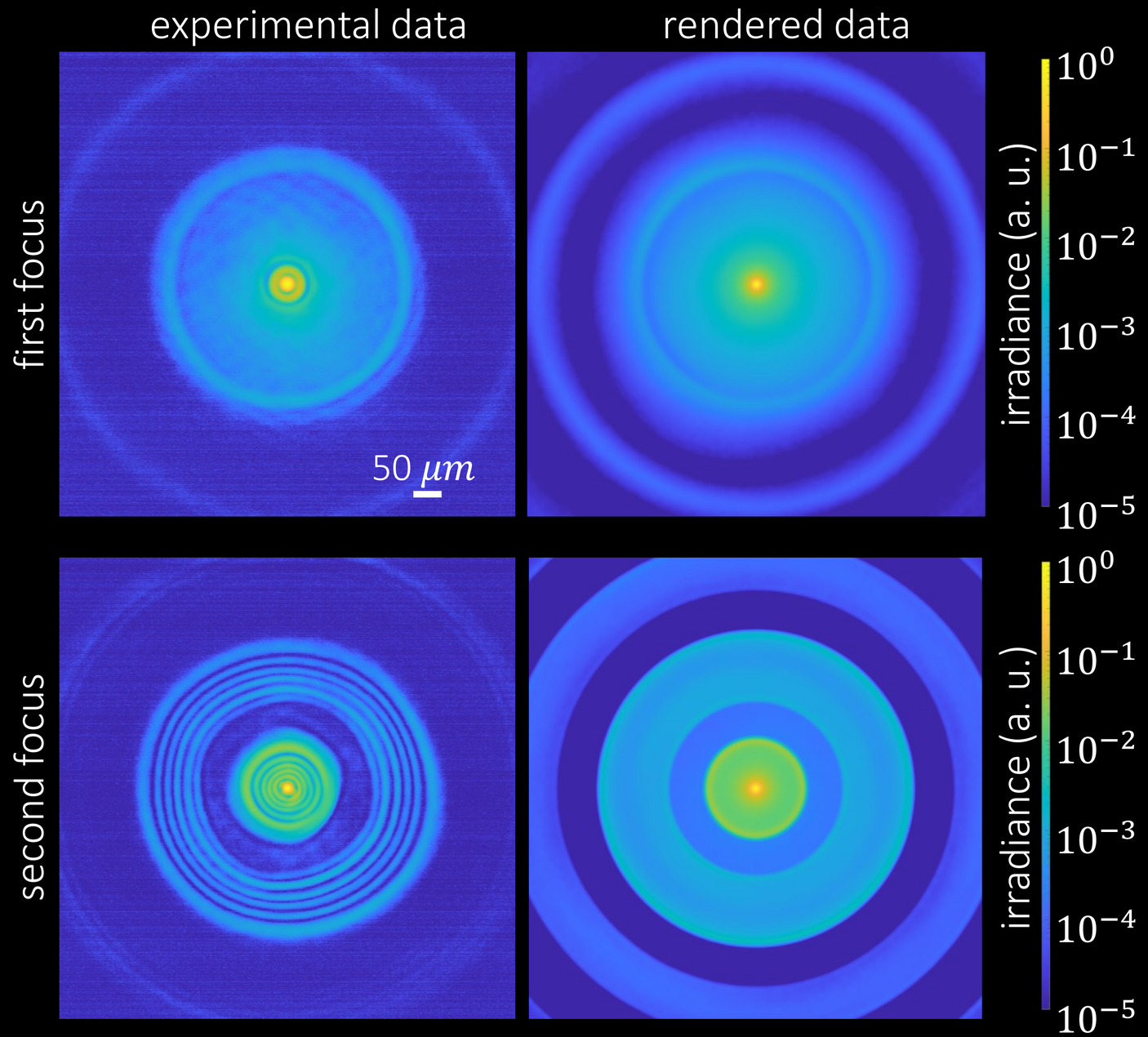
first focus



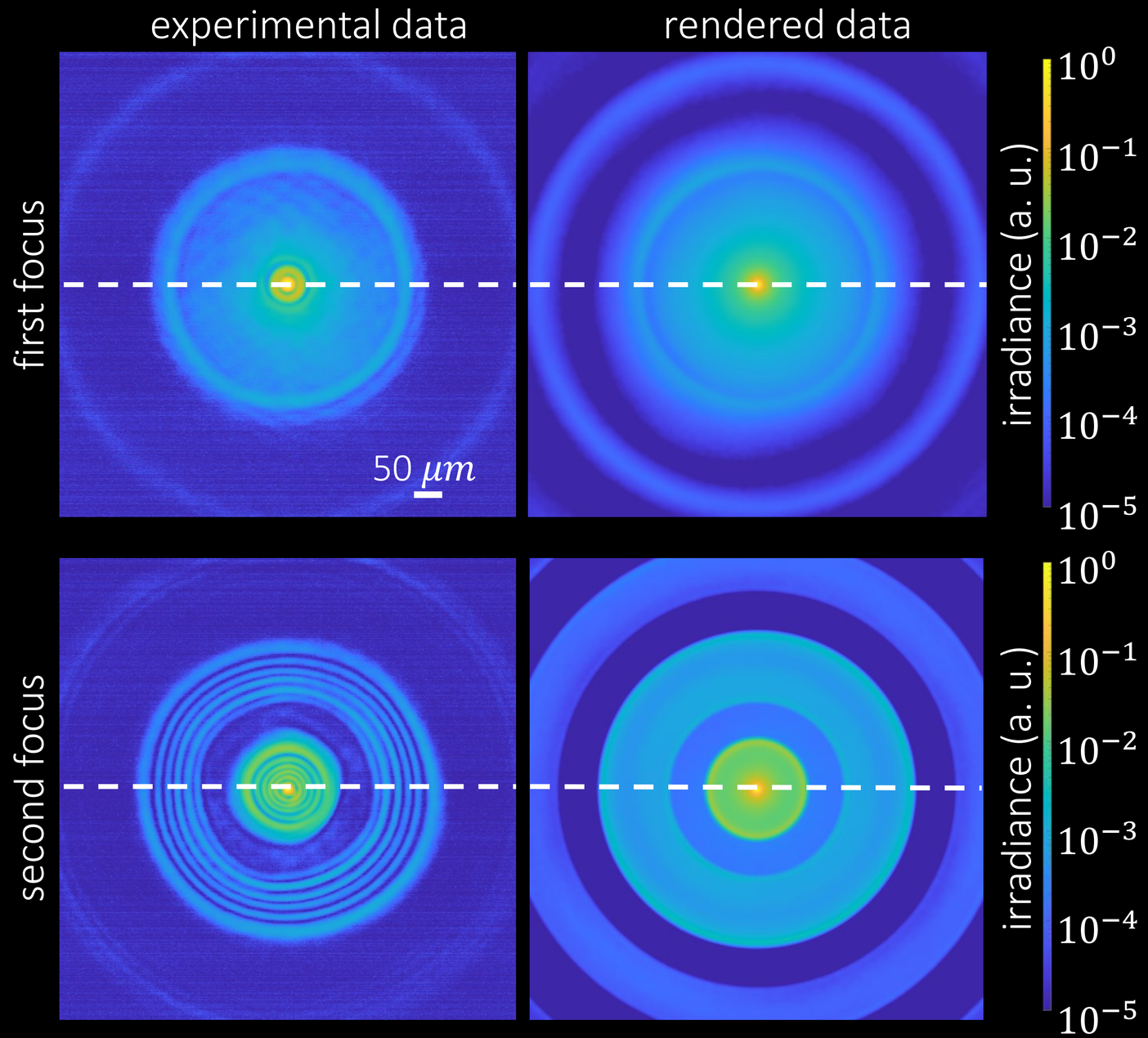
second focus

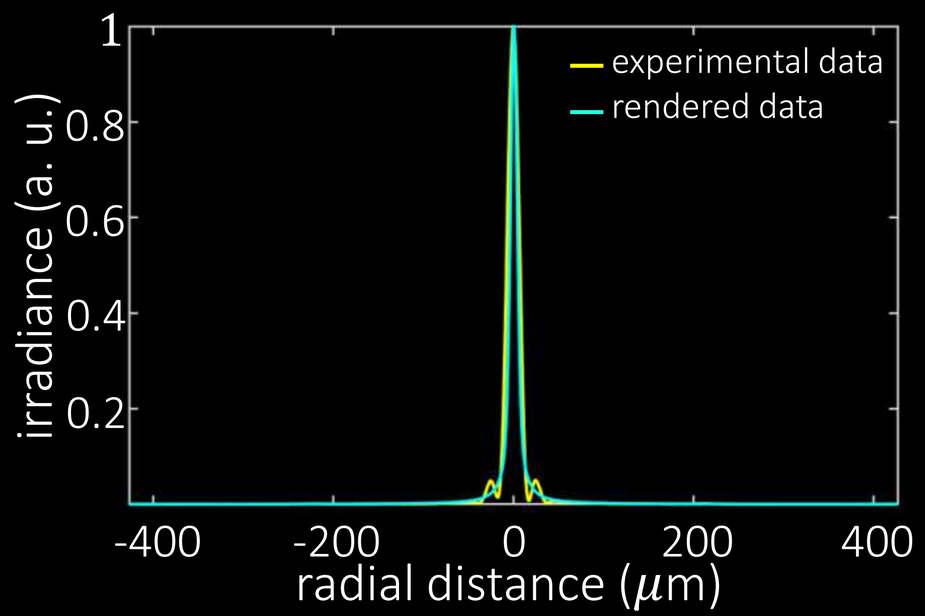


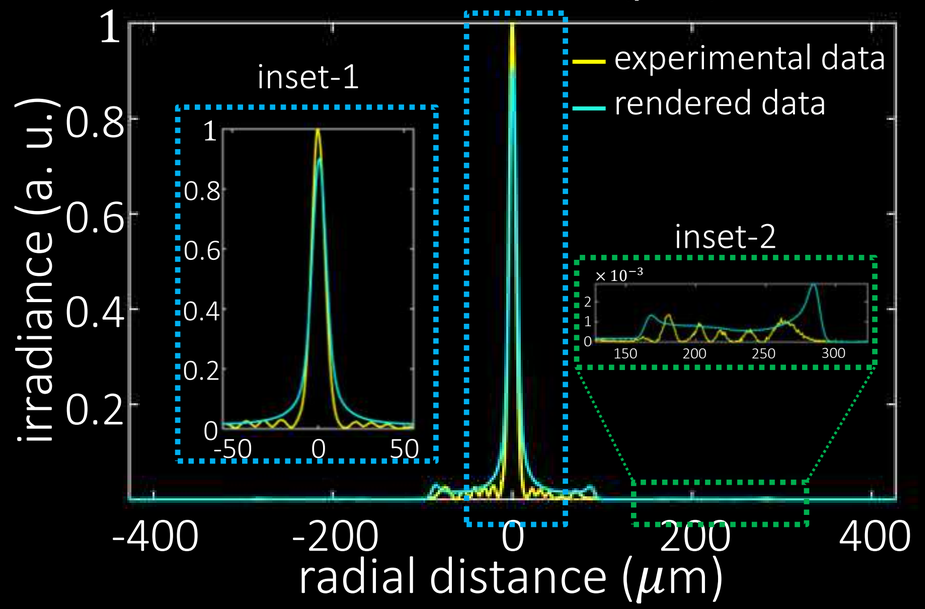
Validation of simulated data



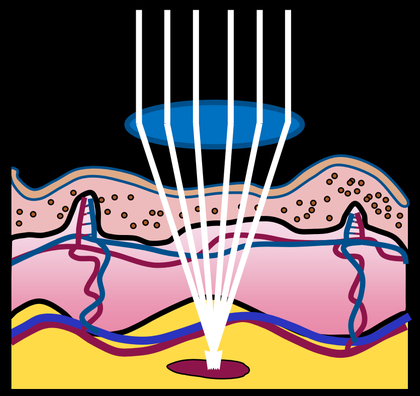
Validation of simulated data





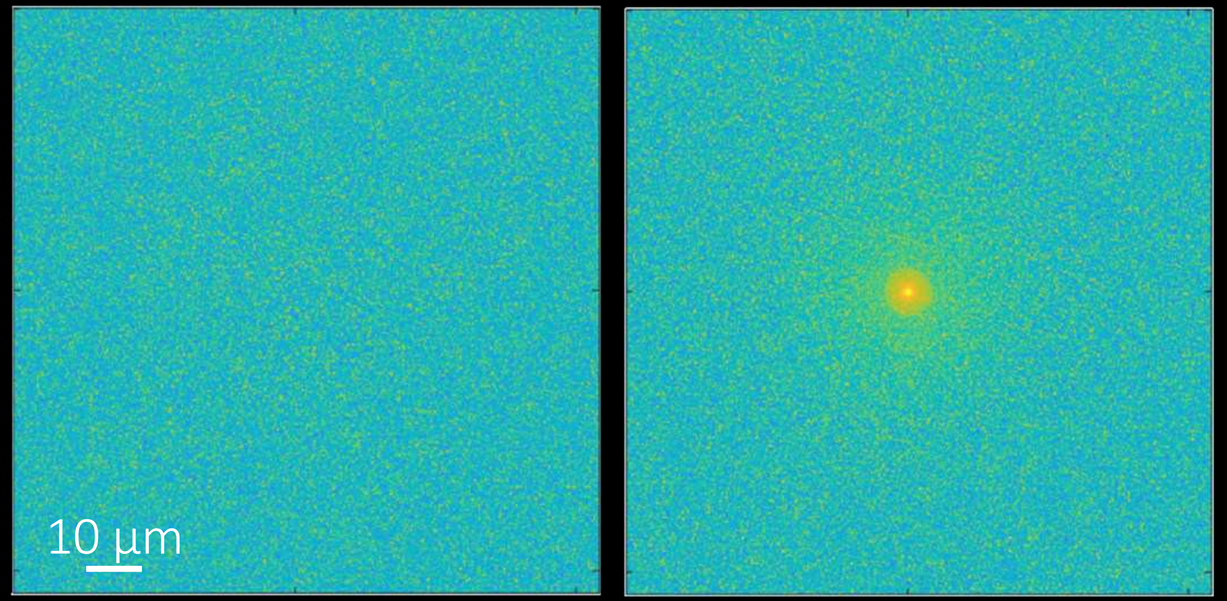


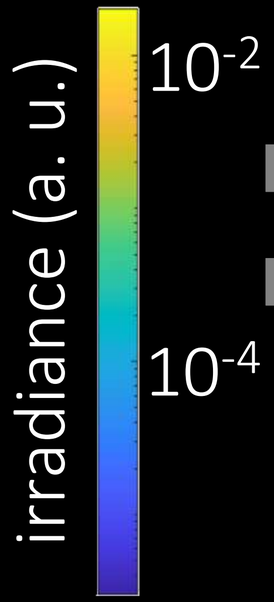
Optimized configurations are better than ideal external lens

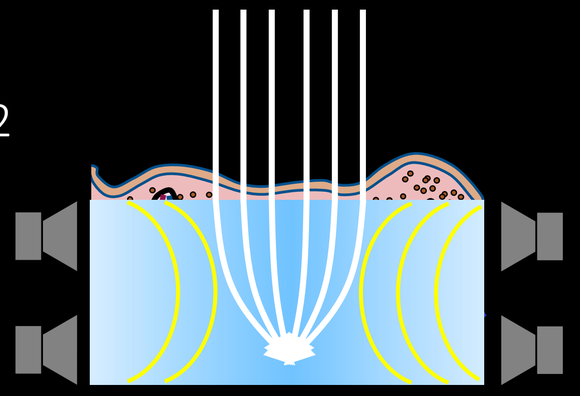


ideal external lens

ultrasound







On human bladder (10 scattering lengths, 2.67 mm thick)

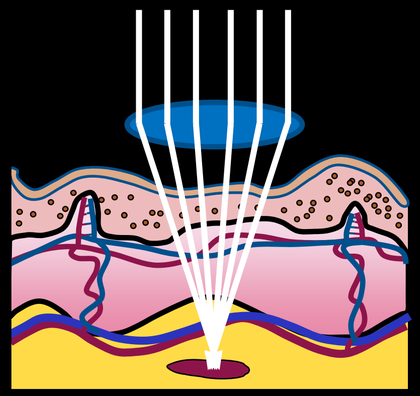
- 50% higher focusing performance than external lens.
- 300% higher focusing performance than previous designs.

On brain tissue (50 scattering lengths, 7.5 mm thick)

- 15% higher focusing performance than external lens.
- Experimentally validated on tissue phantoms.

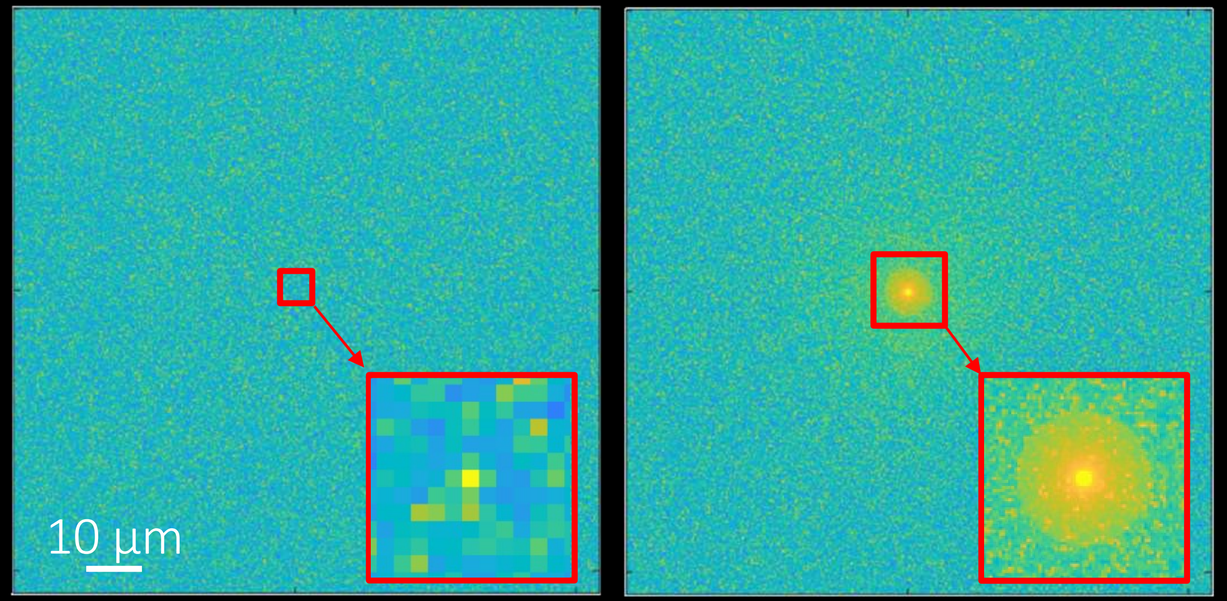
[Pediredla et al., to appear in Nature Communications]

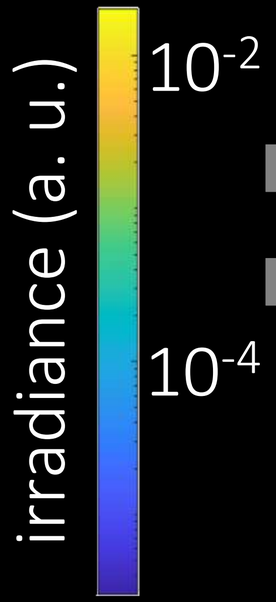
Optimized configurations are better than ideal external lens

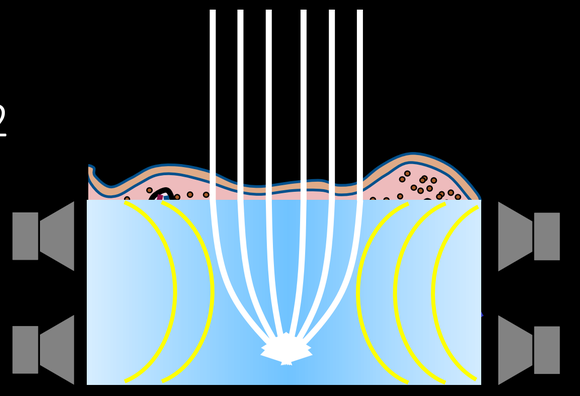


ideal external lens

ultrasound







On human bladder (10 scattering lengths, 2.67 mm thick)

- 50% higher focusing performance than external lens.
- 300% higher focusing performance than previous designs.

On brain tissue (50 scattering lengths, 7.5 mm thick)

- 15% higher focusing performance than external lens.
- Experimentally validated on tissue phantoms.

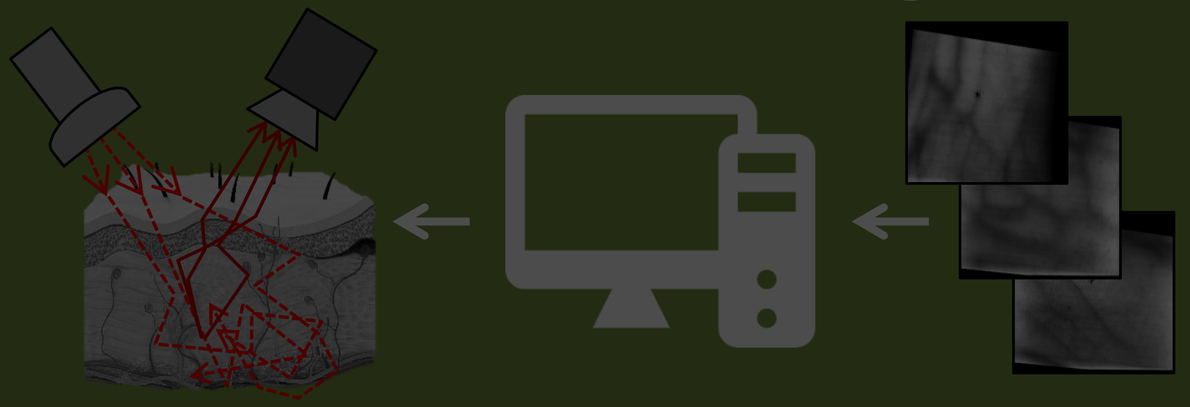
[Pediredla et al., to appear in Nature Communications]

Physics-based rendering and its applications to computational imaging

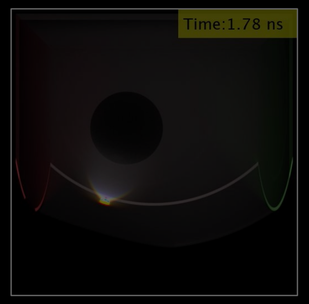
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

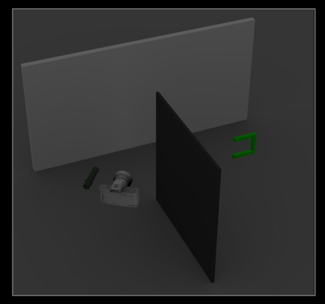
inverse rendering



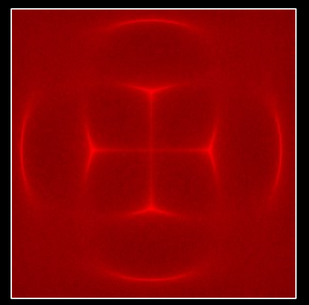
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



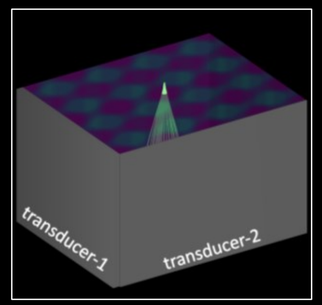
time-of-flight imaging



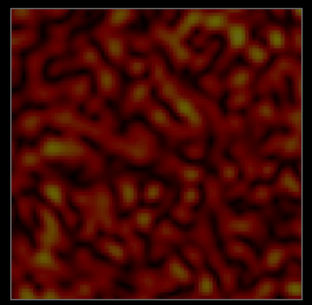
non-line-of-sight imaging



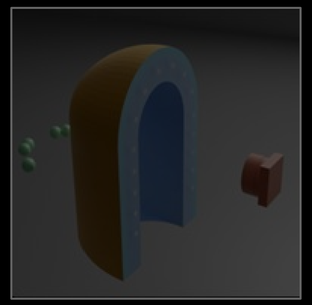
acousto-optic lensing



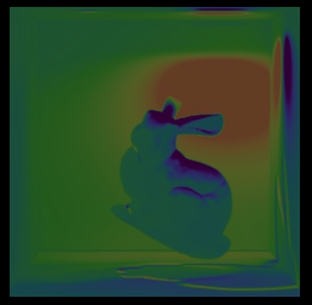
ultrafast light scanning



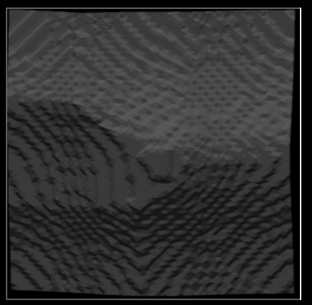
speckle imaging



tactile sensor design

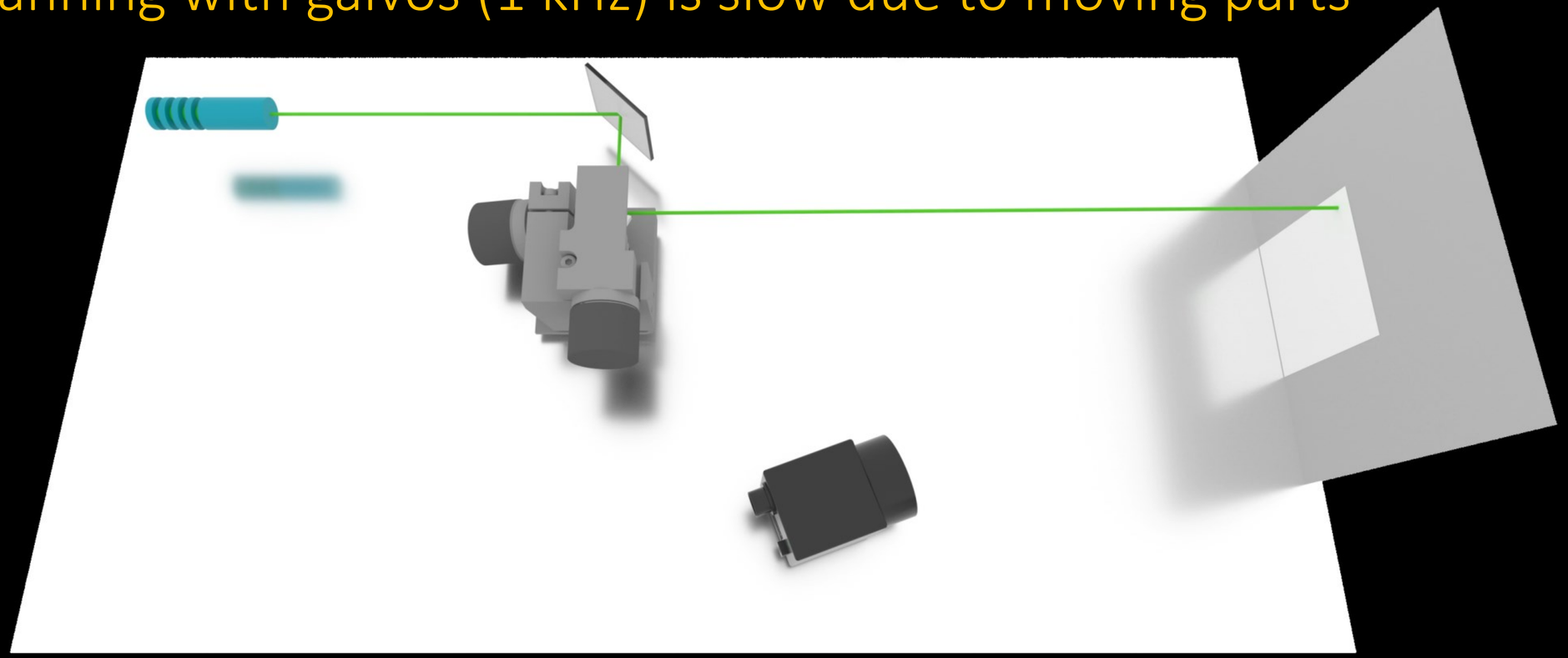


differentiable renderer



inverse problems

scanning with galvos (1 kHz) is slow due to moving parts

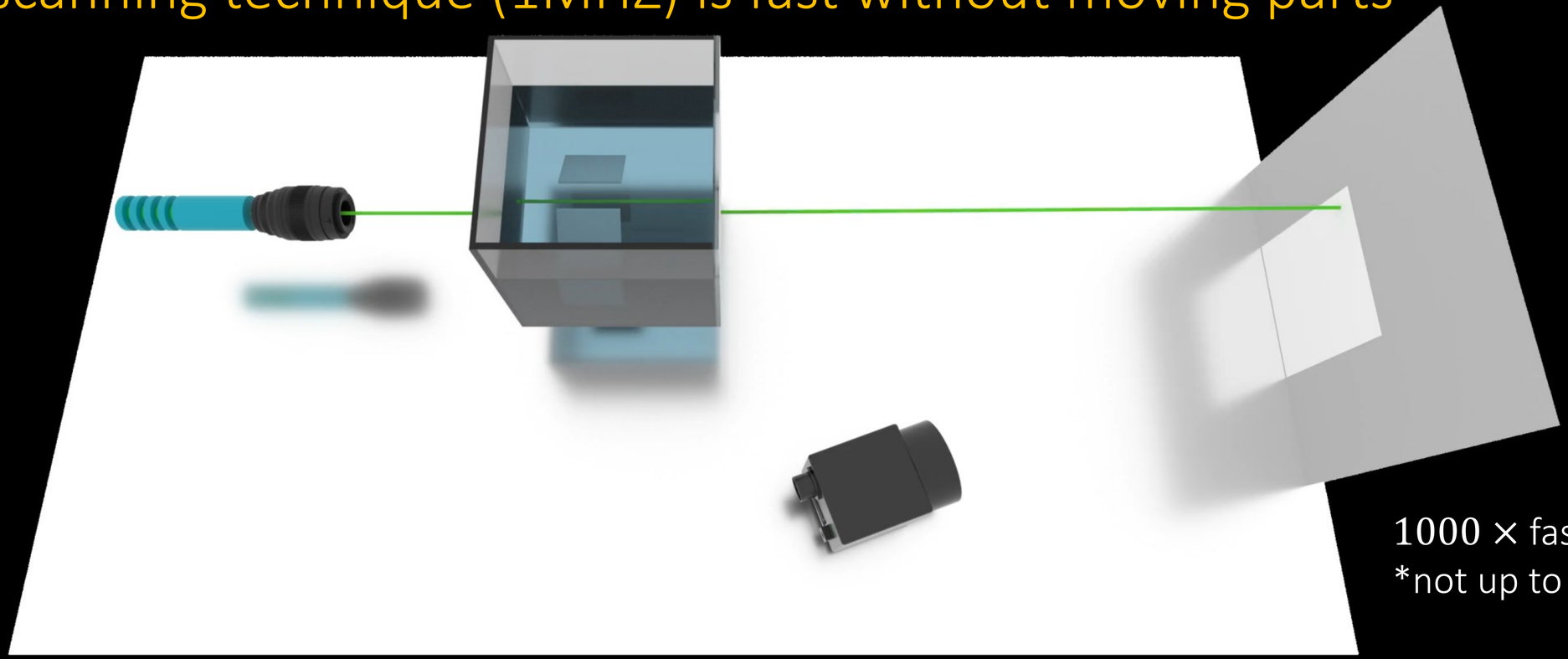


projector

microscopy

lidar

our scanning technique (1MHZ) is fast without moving parts



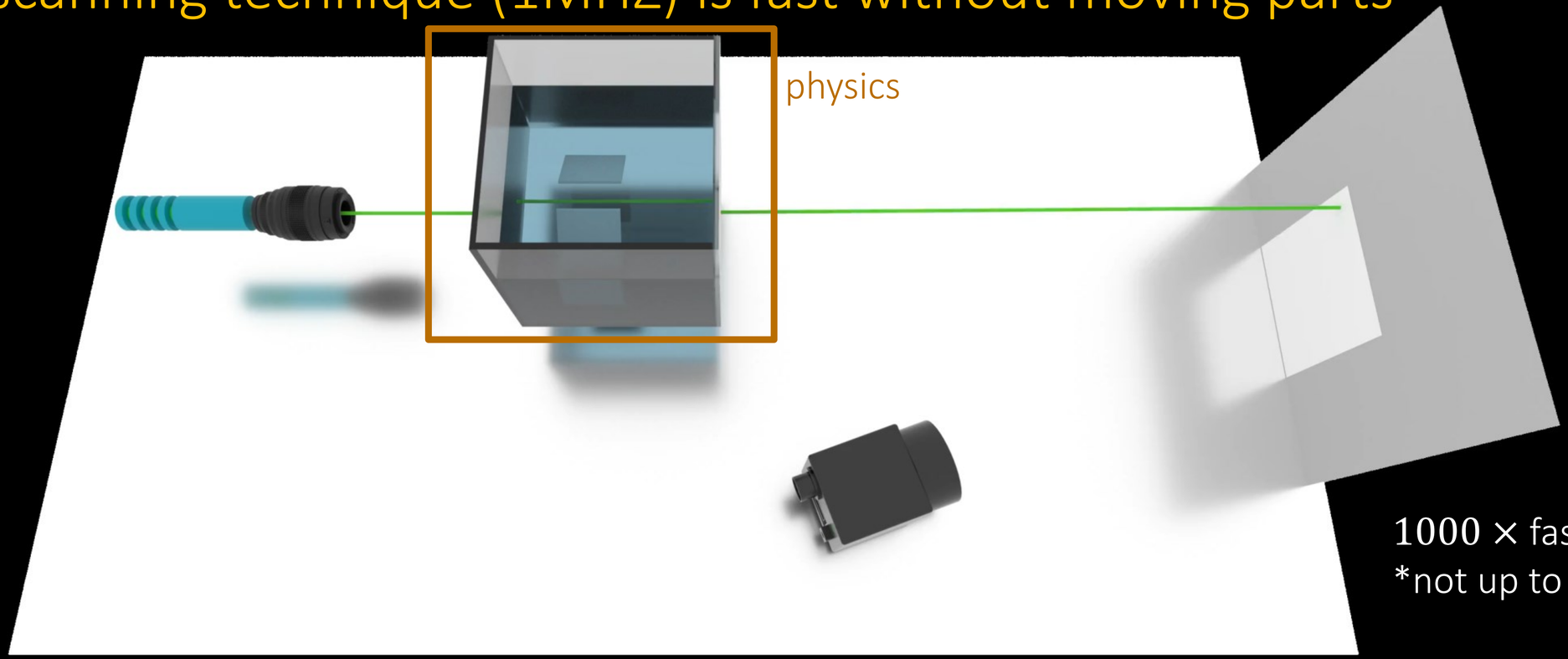
$1000 \times$ faster
*not up to scale

projector

microscopy

lidar

our scanning technique (1MHZ) is fast without moving parts

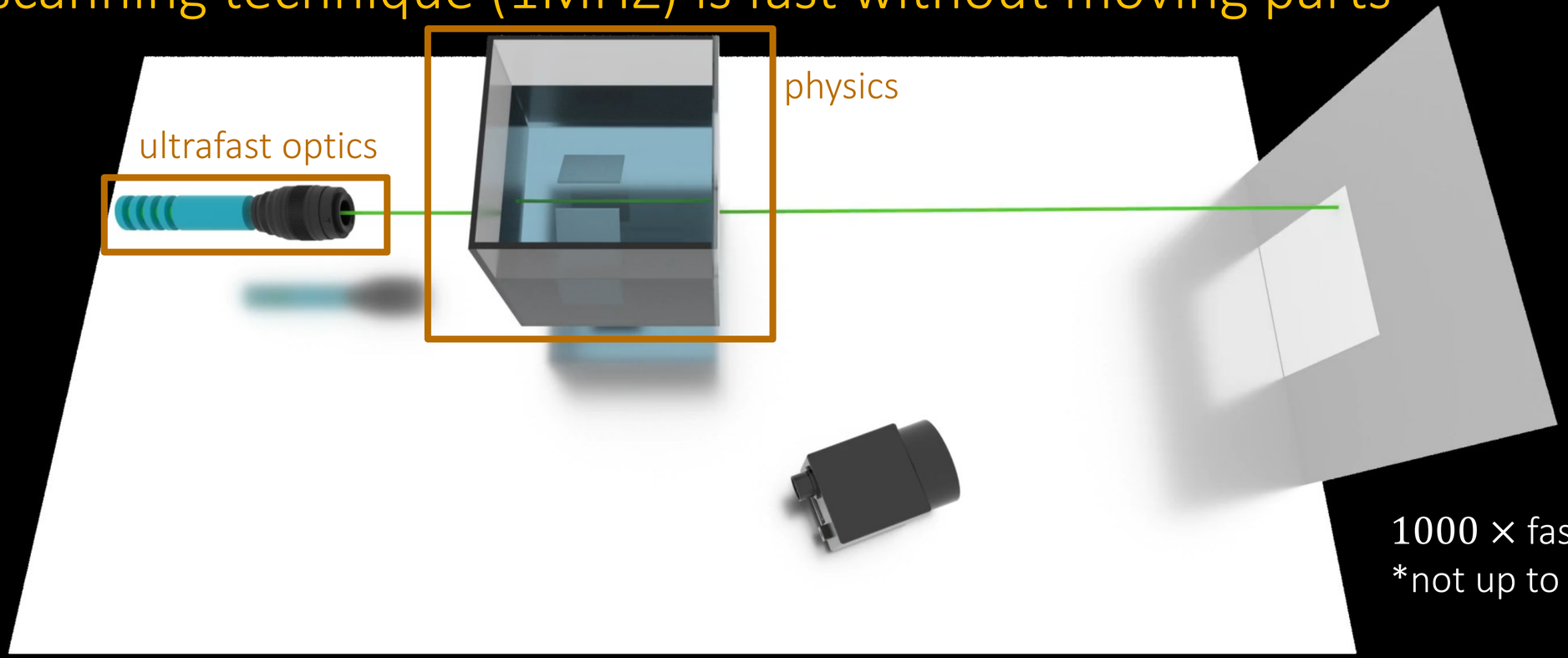


projector

microscopy

lidar

our scanning technique (1MHZ) is fast without moving parts

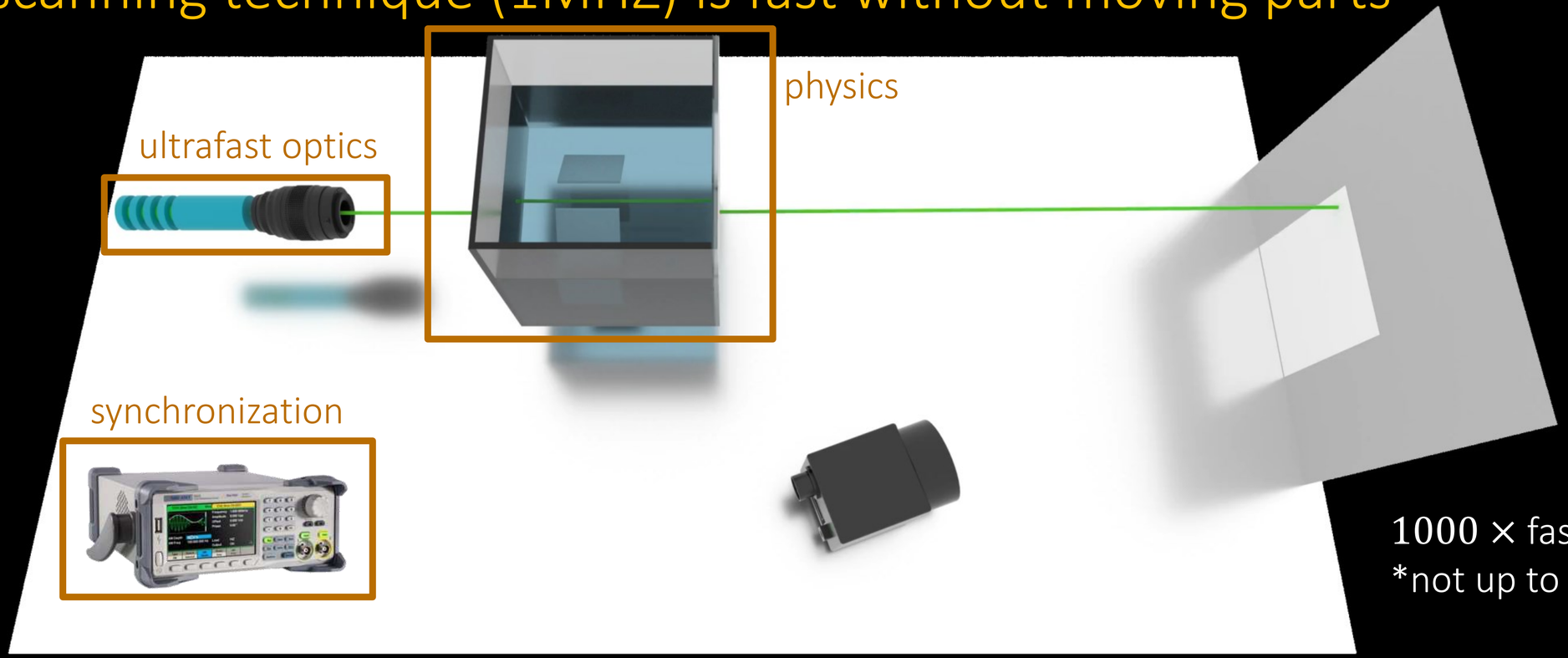


projector

microscopy

lidar

our scanning technique (1MHZ) is fast without moving parts

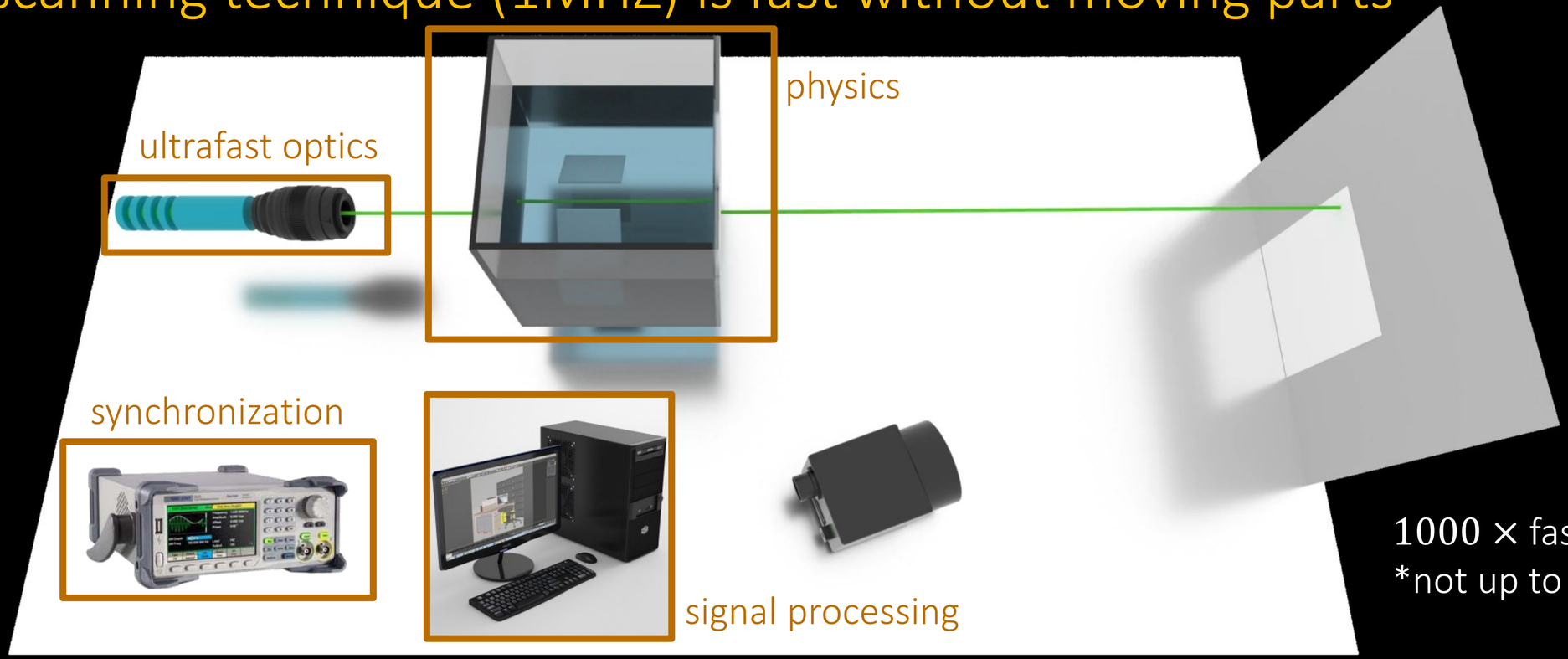


projector

microscopy

lidar

our scanning technique (1MHZ) is fast without moving parts

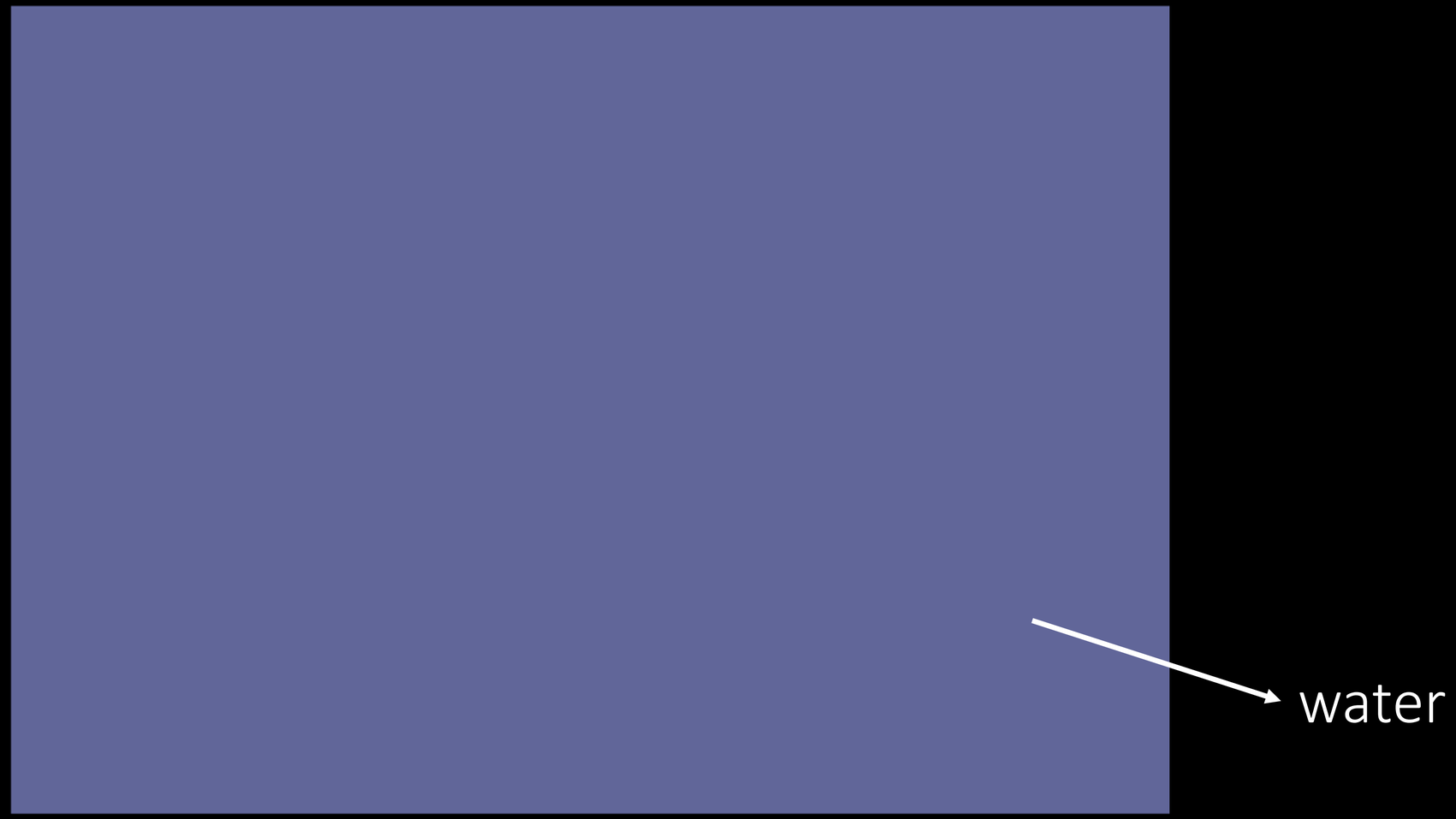


projector

microscopy

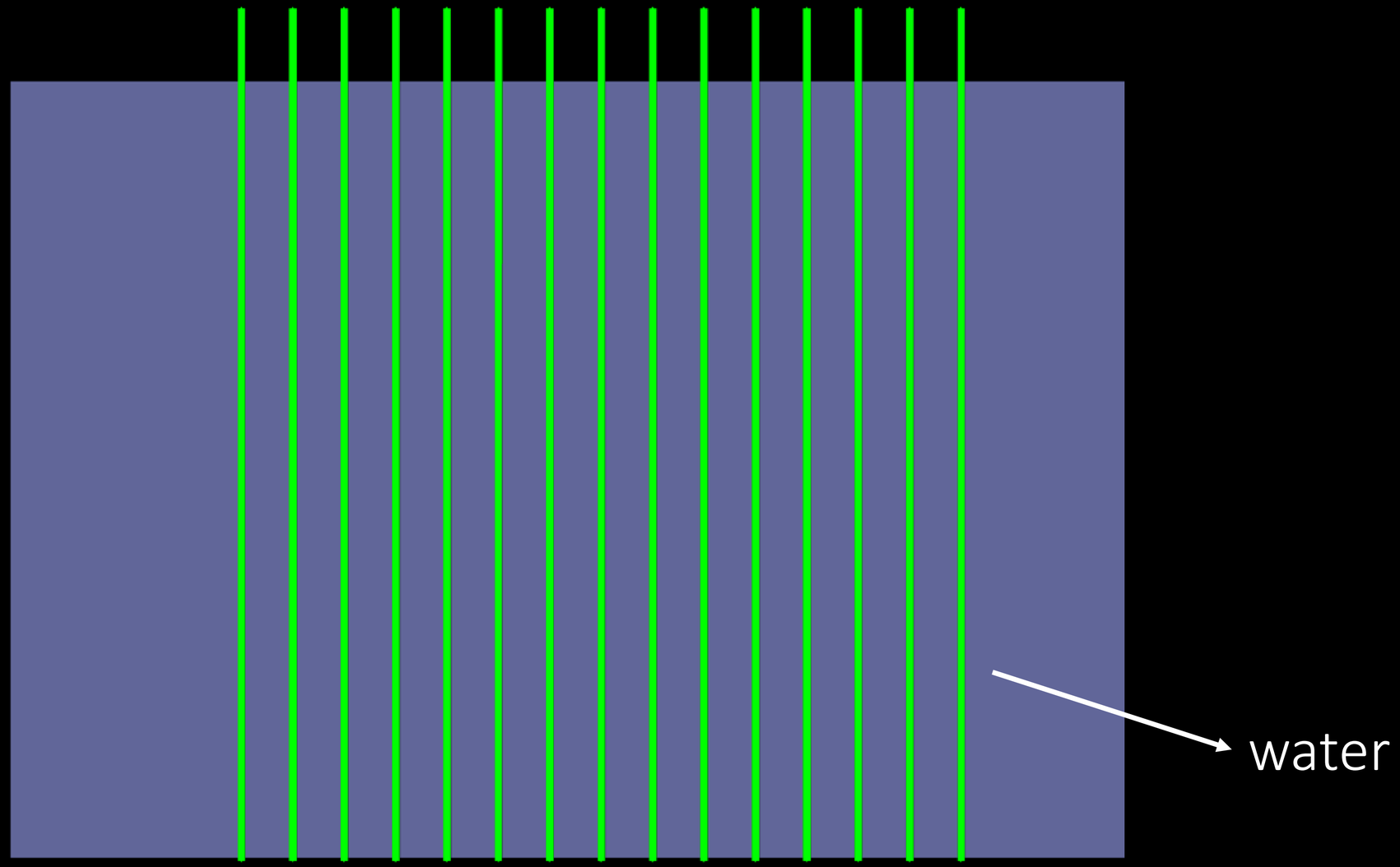
lidar

Physics sound, light, and matter interaction



2D visualization

Physics sound, light, and matter interaction



2D visualization

Physics sound, light, and matter interaction

2D visualization

Physics sound, light, and matter interaction

nm scale

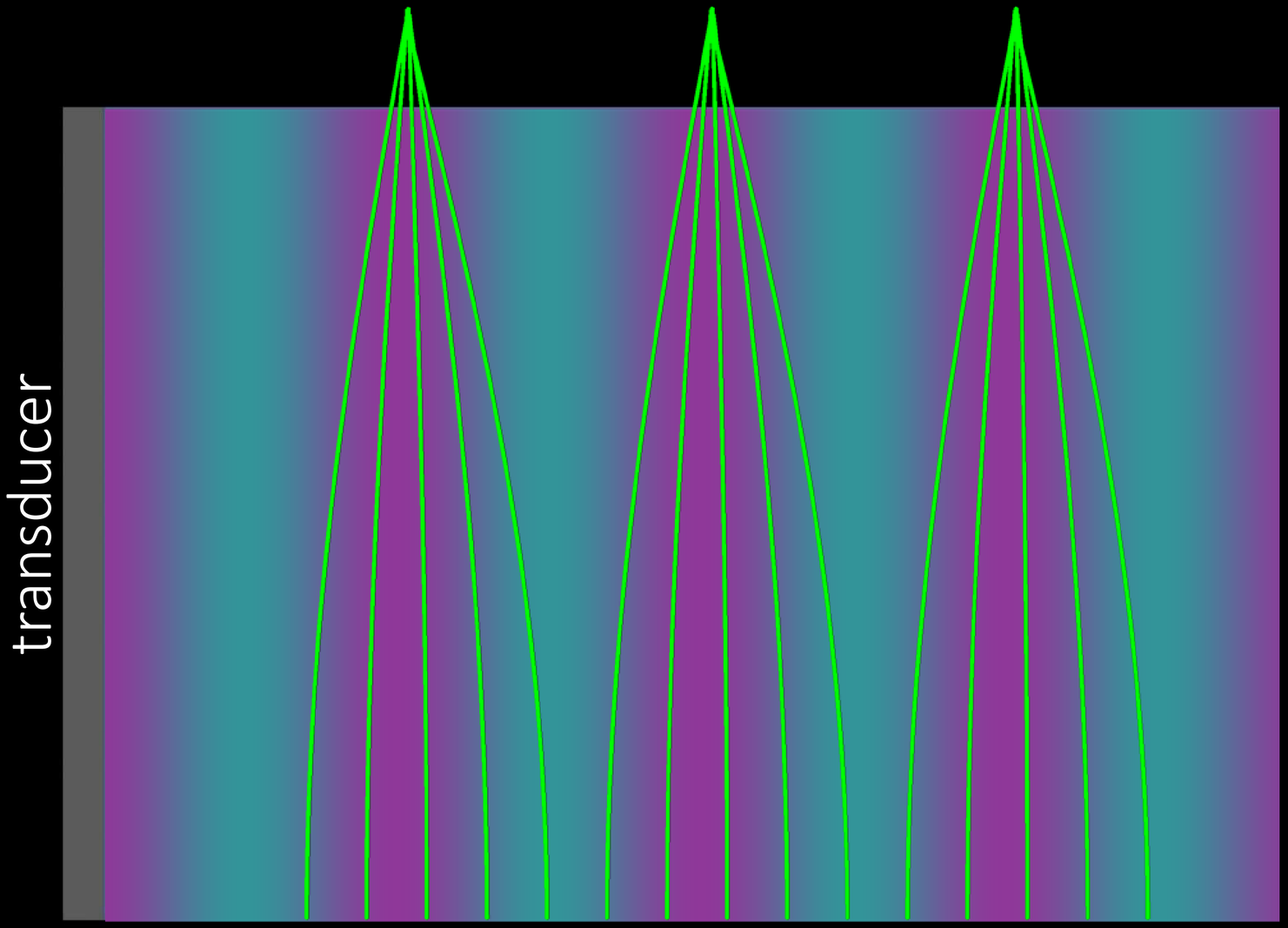
(exaggerated)

2D visualization

Physics sound, light, and matter interaction

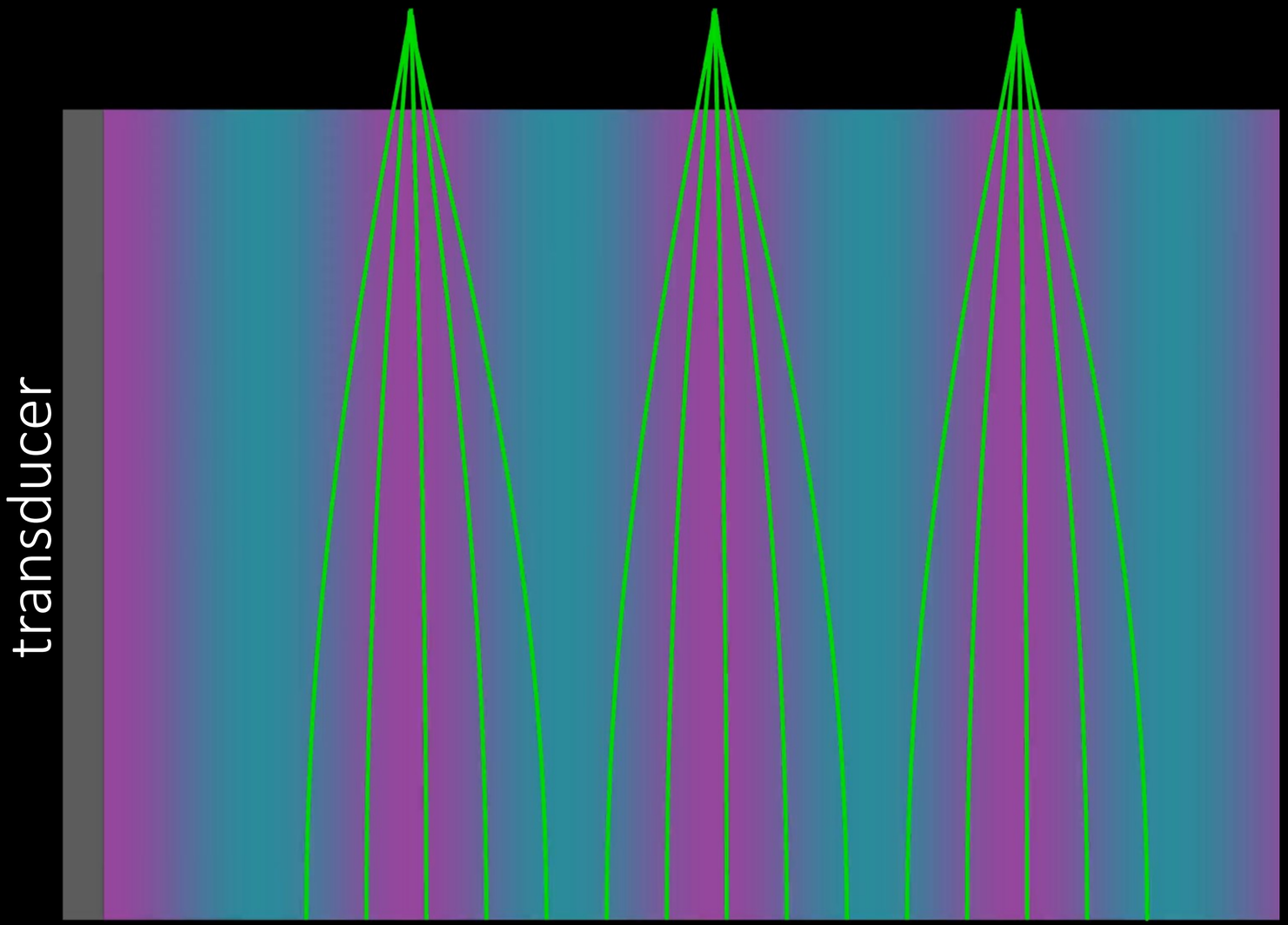
2D visualization

Physics sound, light, and matter interaction



2D visualization

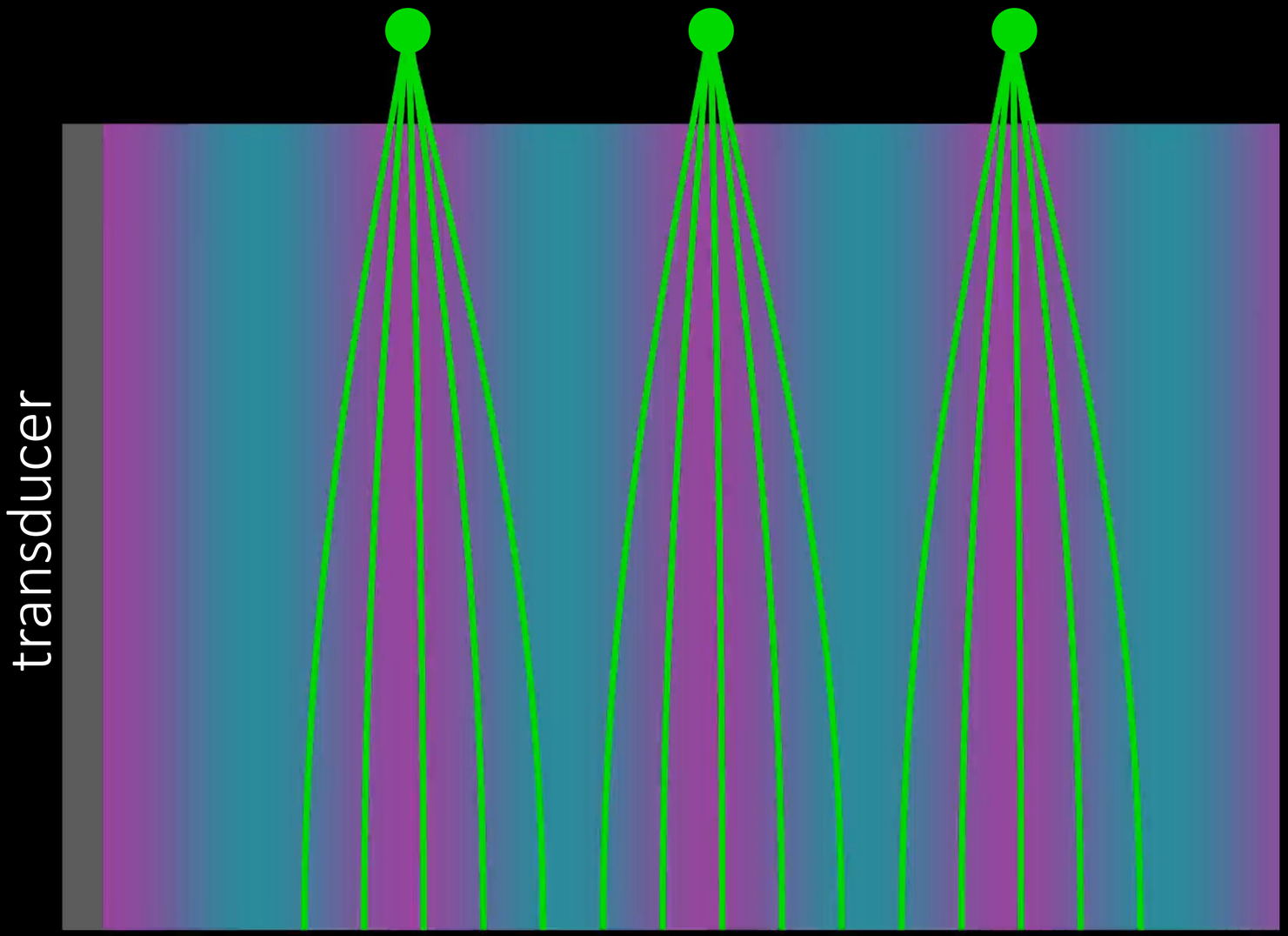
Physics sound, light, and matter interaction



focus travels at the speed of 1.5 kmps

2D visualization

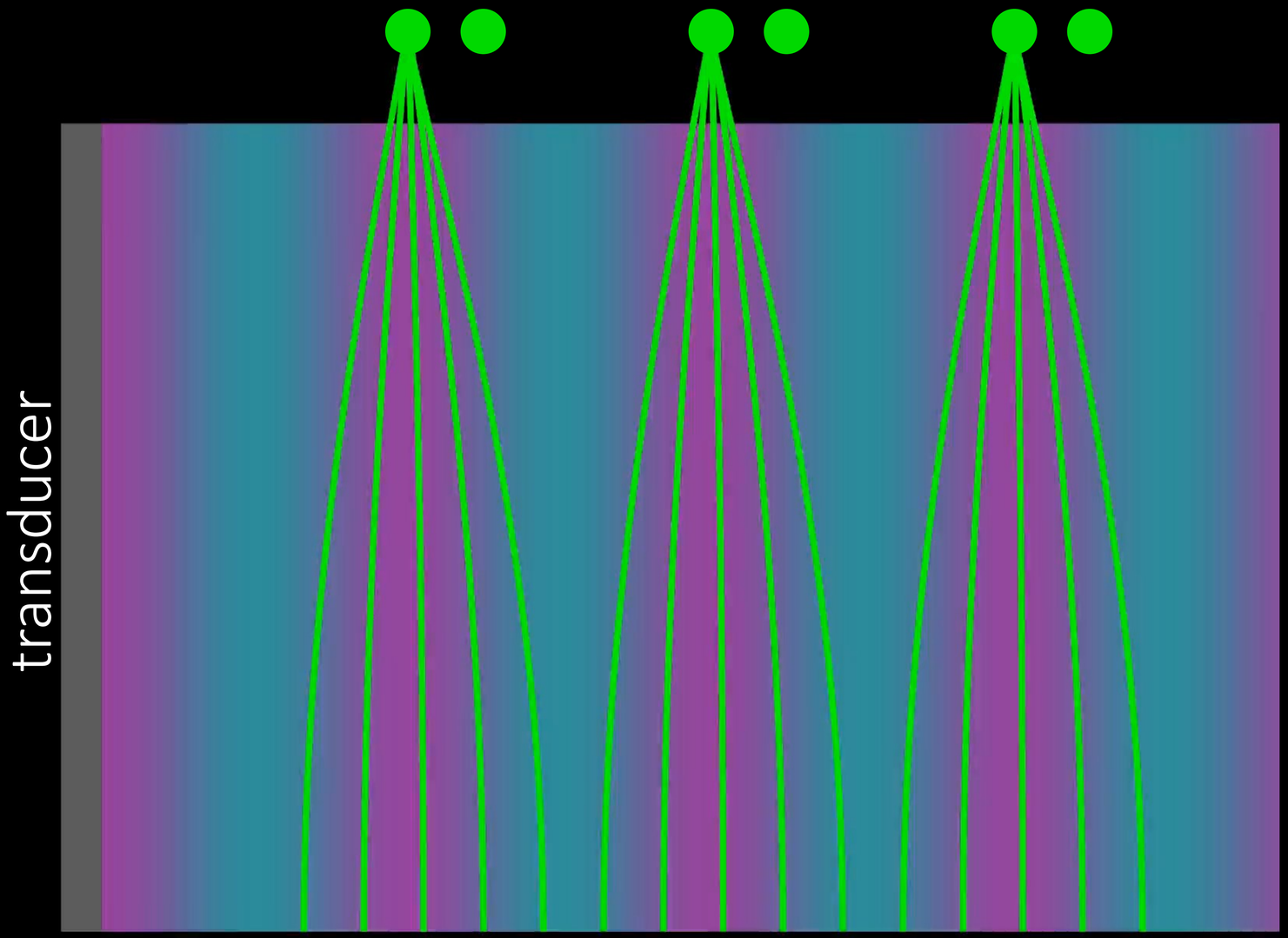
Ultrafast optics and synchronization



ultrafast synchronized optics that illuminates same spot

2D visualization

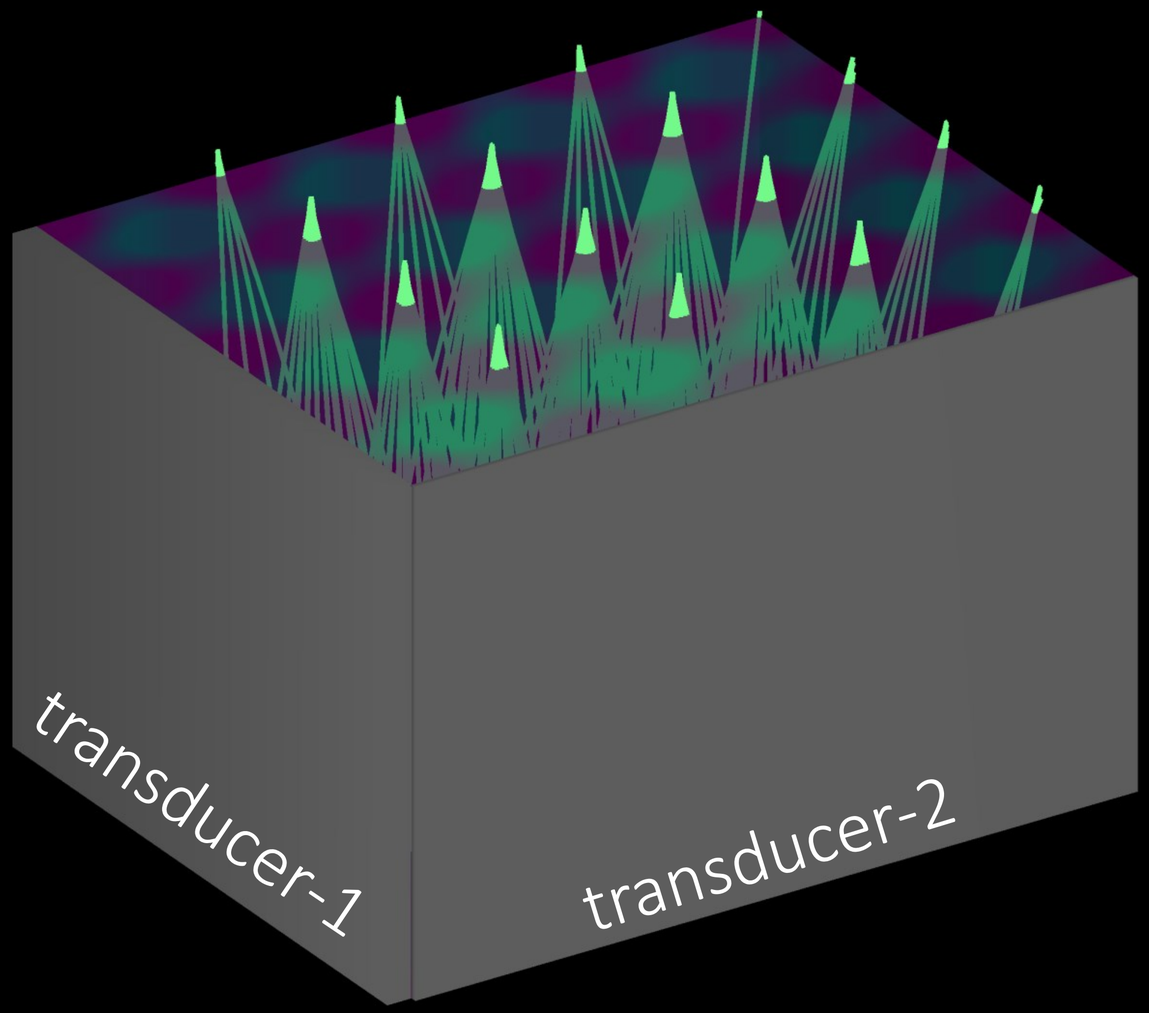
Ultrafast optics and synchronization



illuminating two spots per cycle

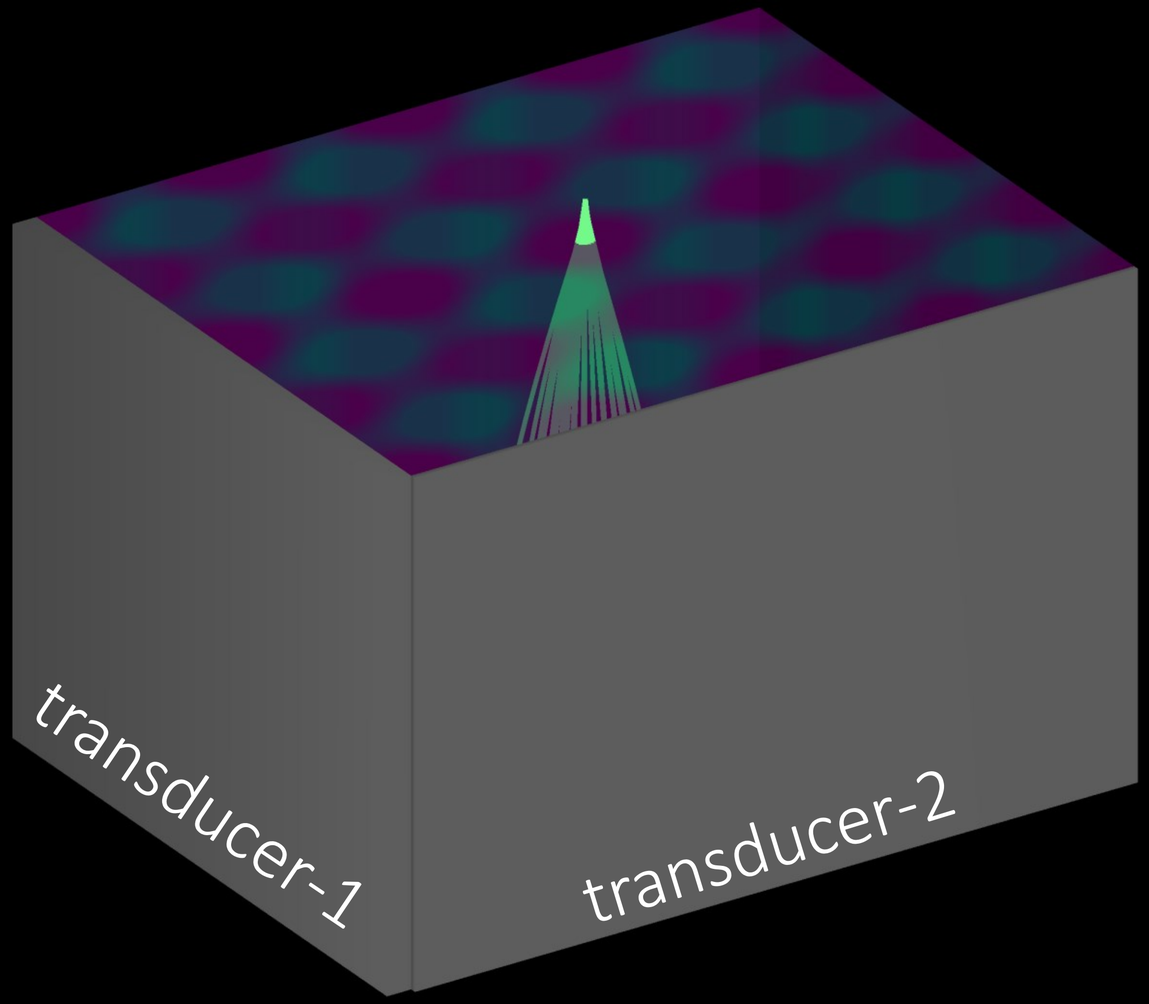
2D visualization

Dot projector



2D visualization

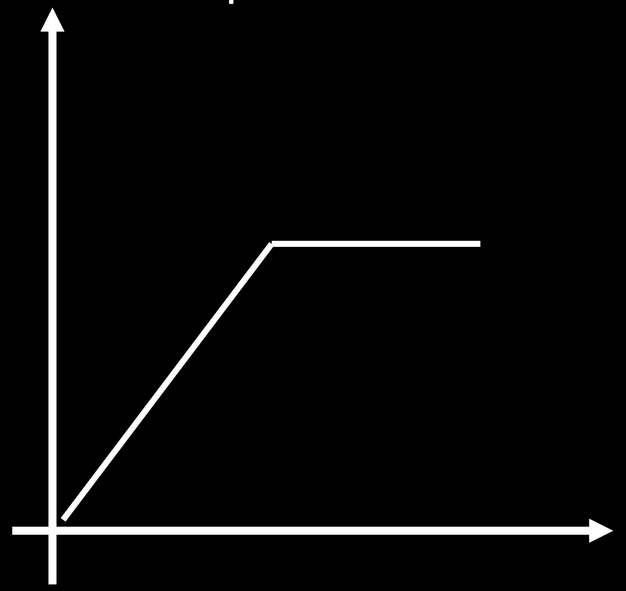
Dot projector

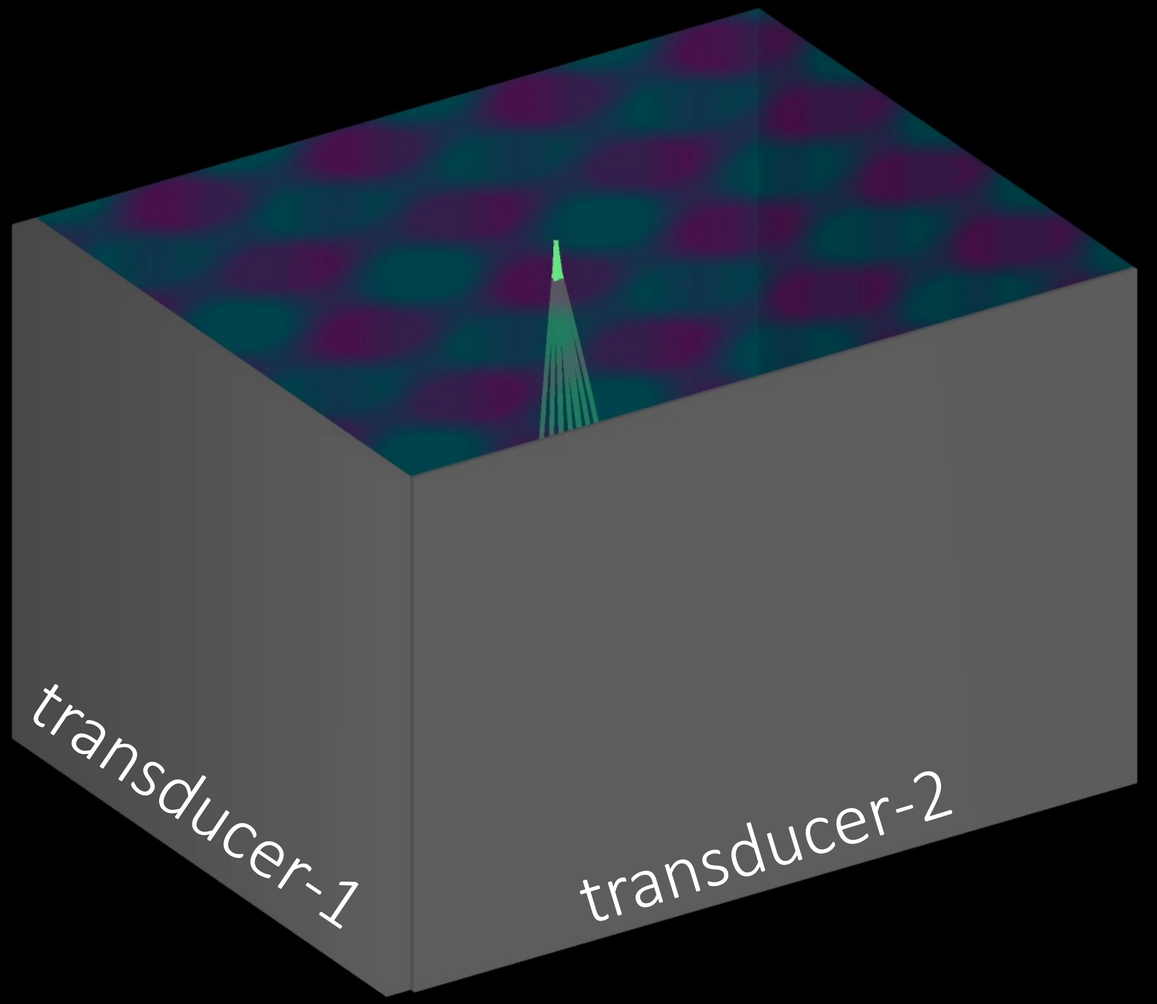


2D visualization

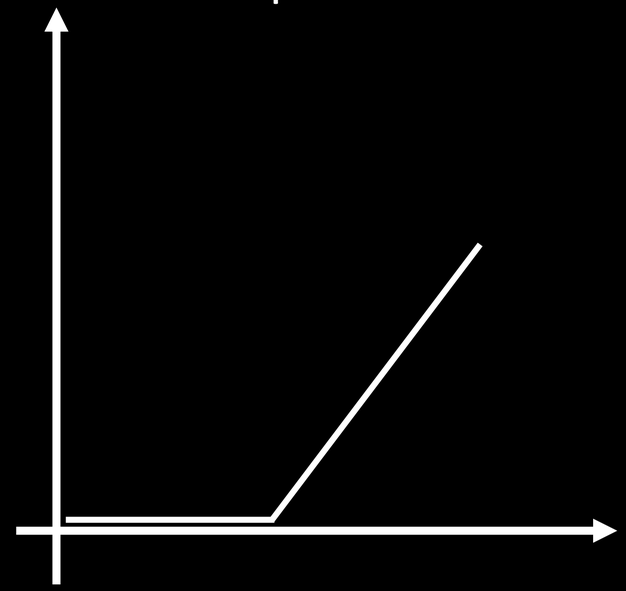
Dot projector

transducer-1
phase

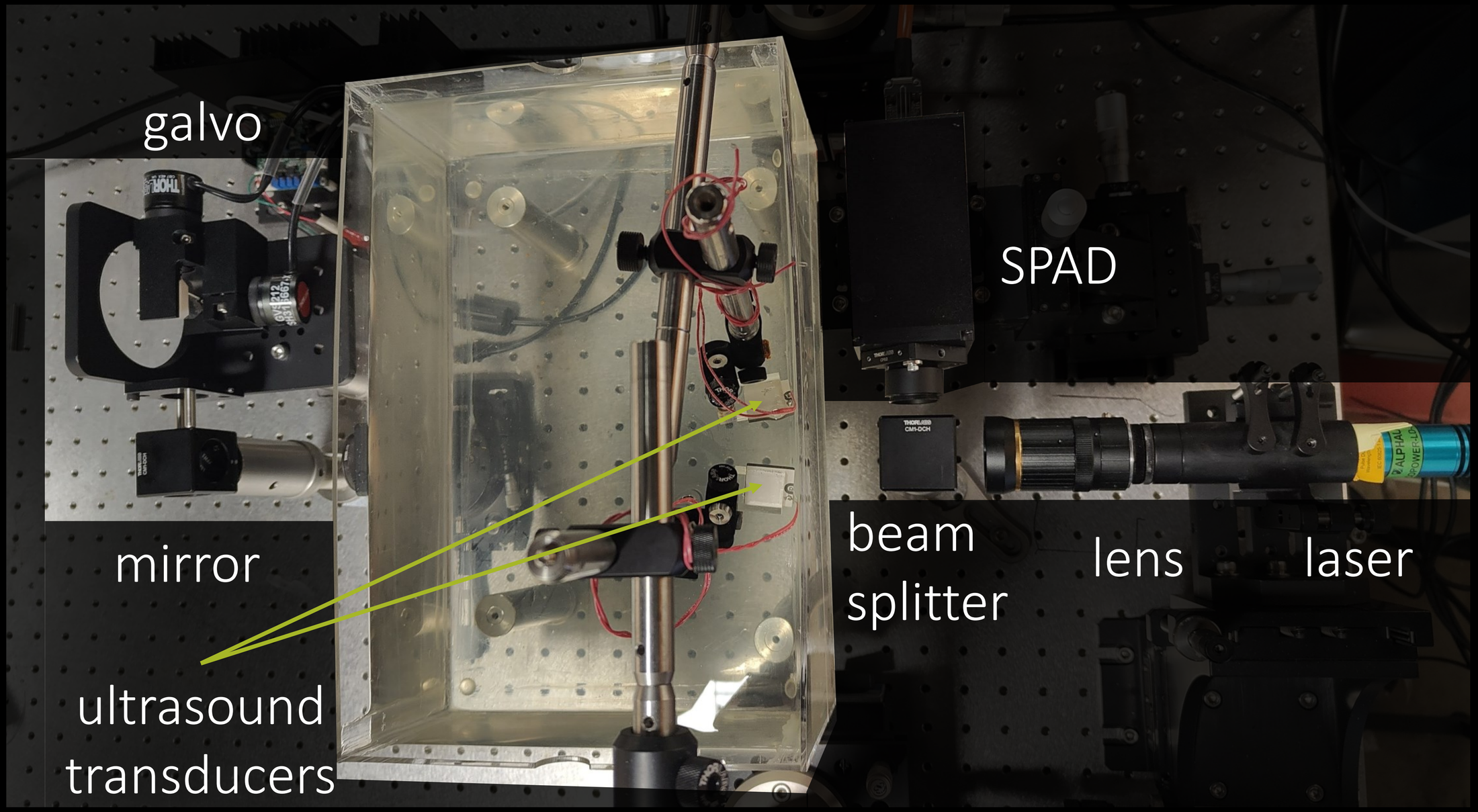




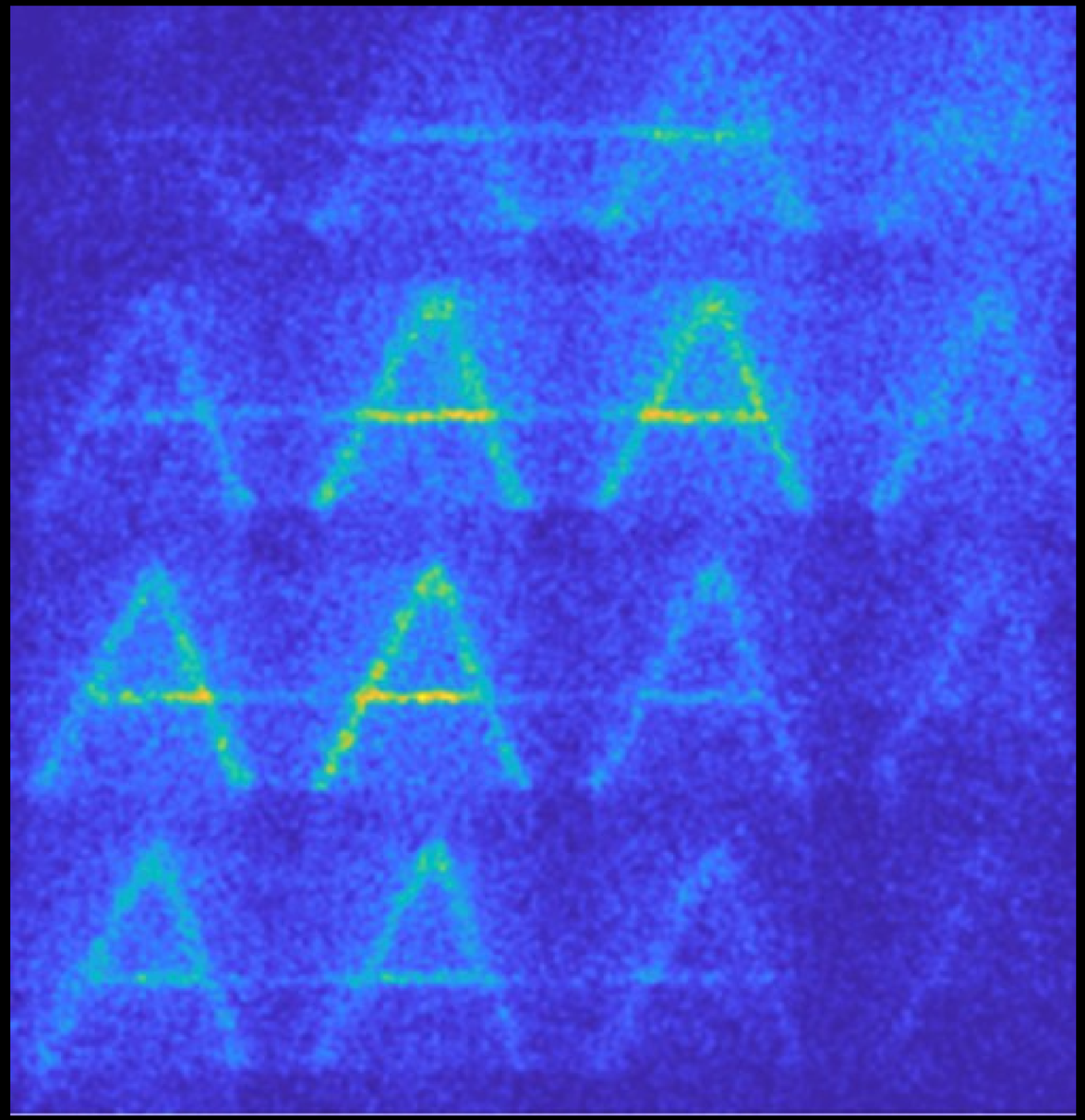
transducer-2
phase

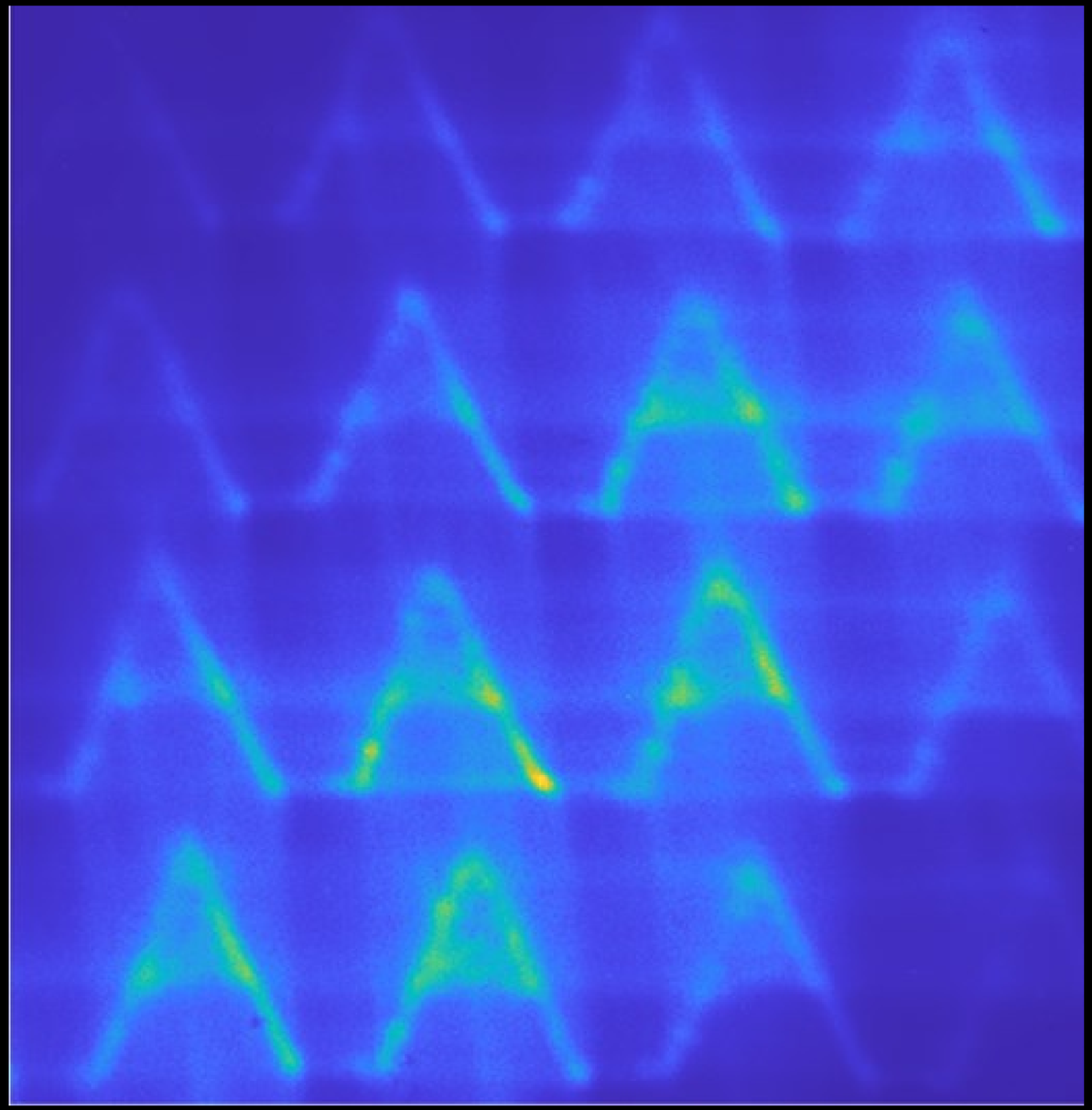


Hardware experiments



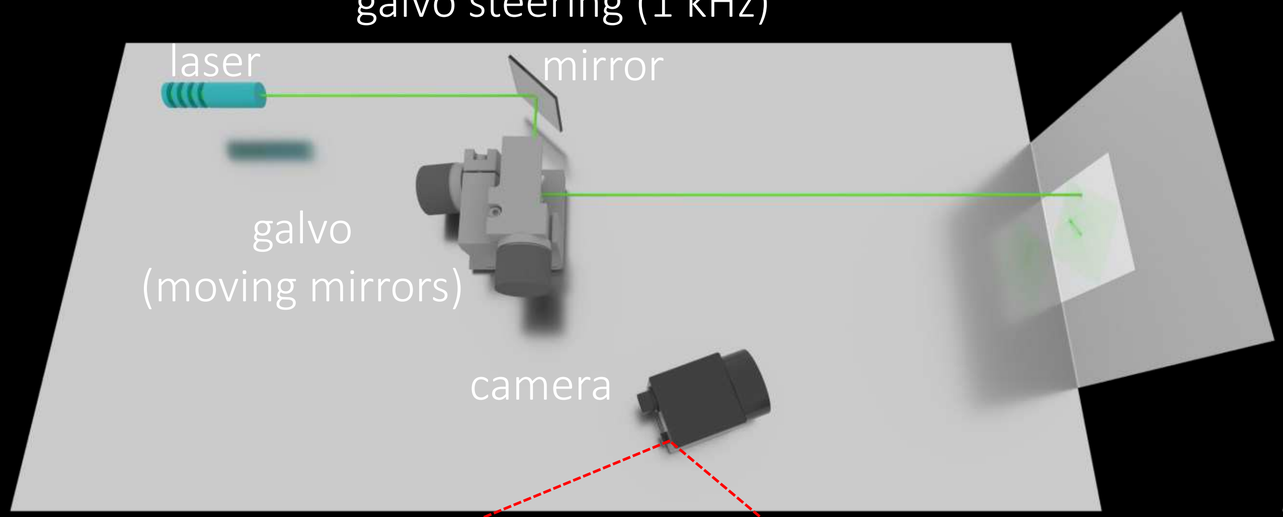
Renderer vs. Hardware



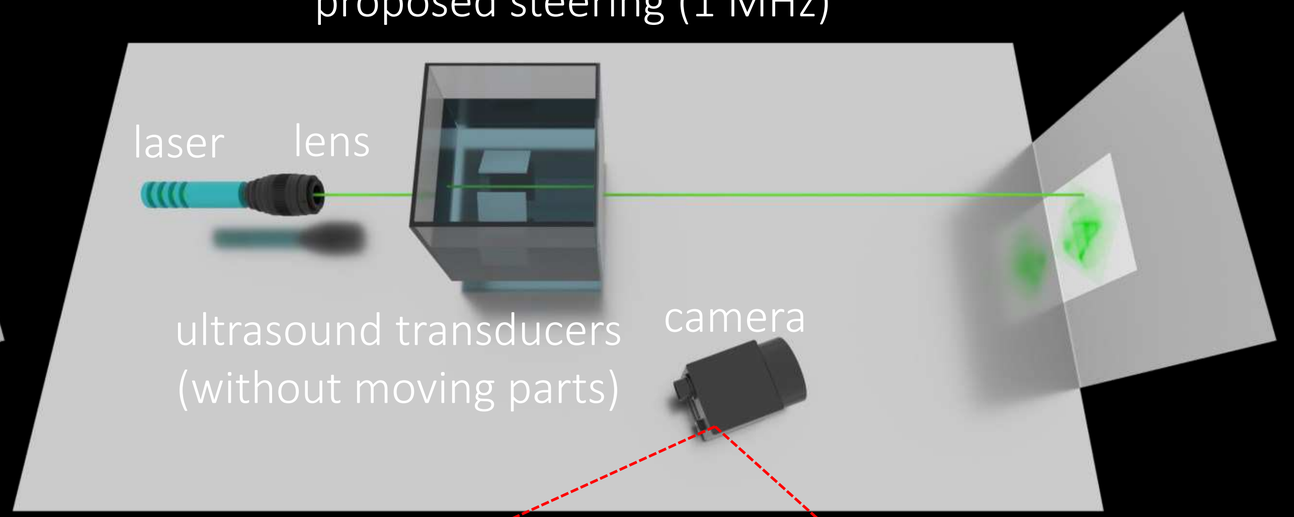


Hardware results: dot projector

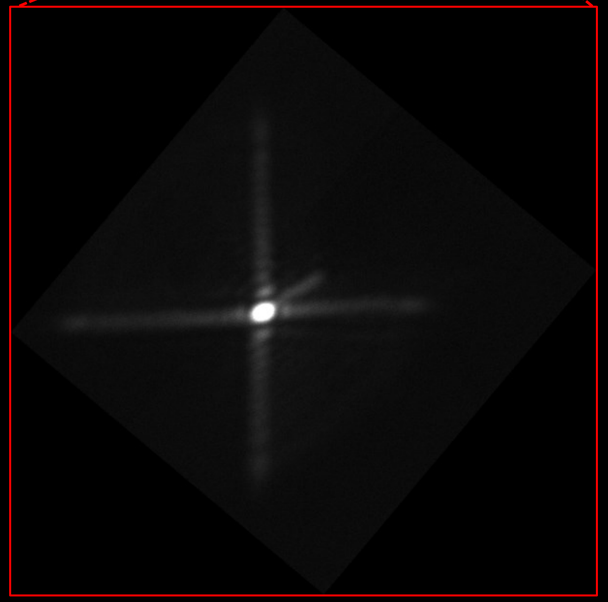
galvo steering (1 kHz)



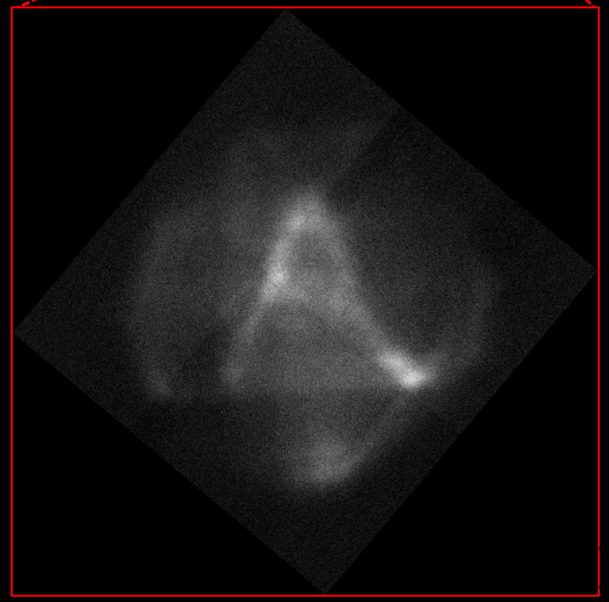
proposed steering (1 MHz)



exposure= 1 ms

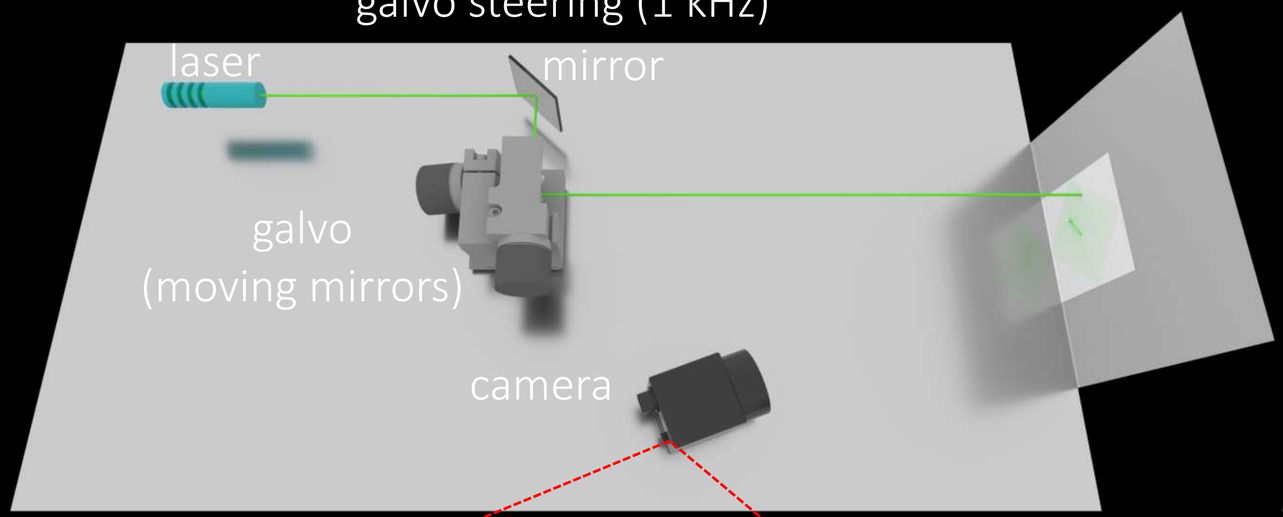


exposure= 1 ms

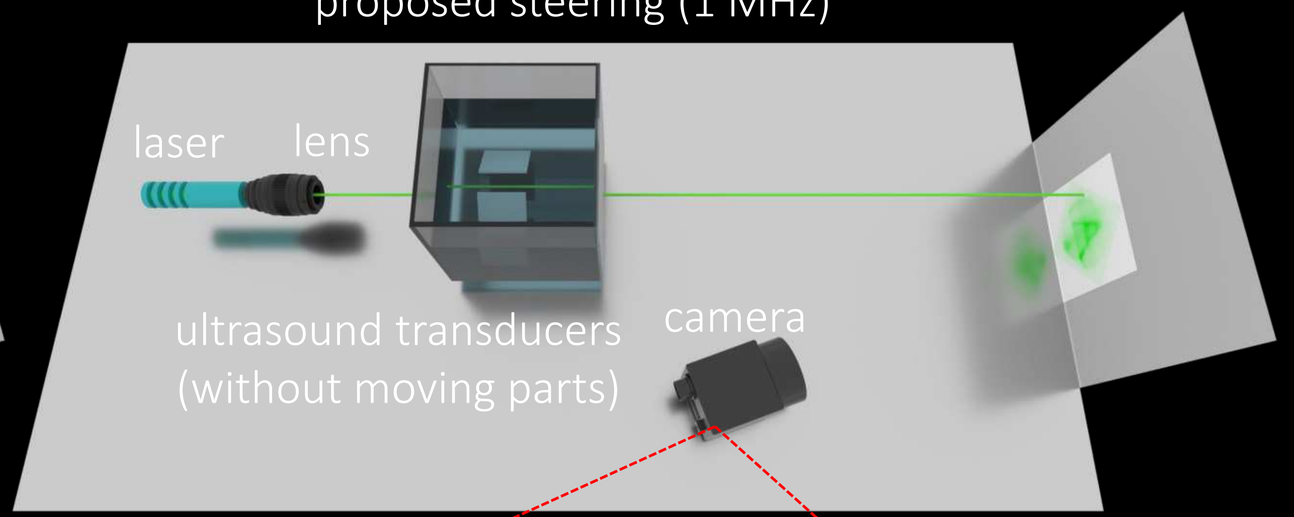


Hardware results: dot projector

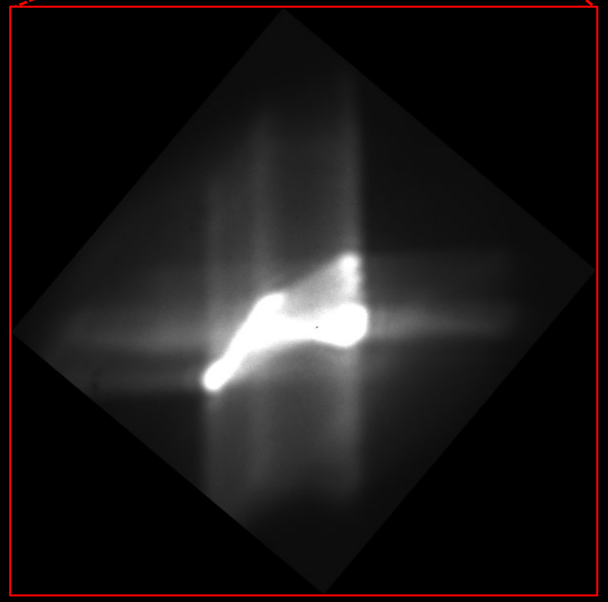
galvo steering (1 kHz)



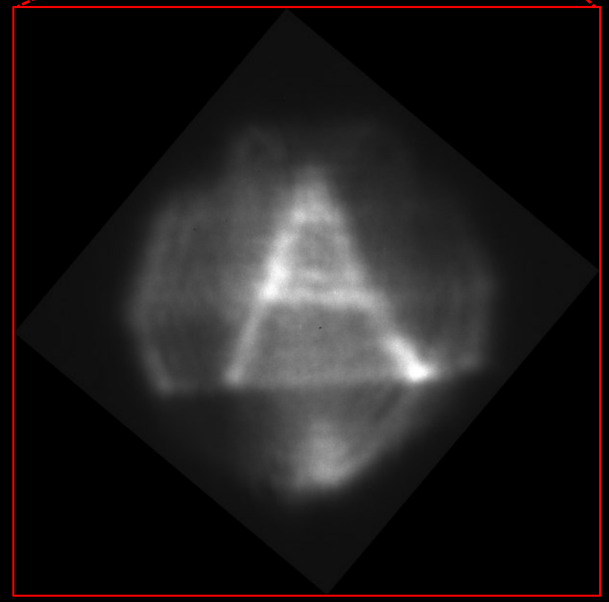
proposed steering (1 MHz)



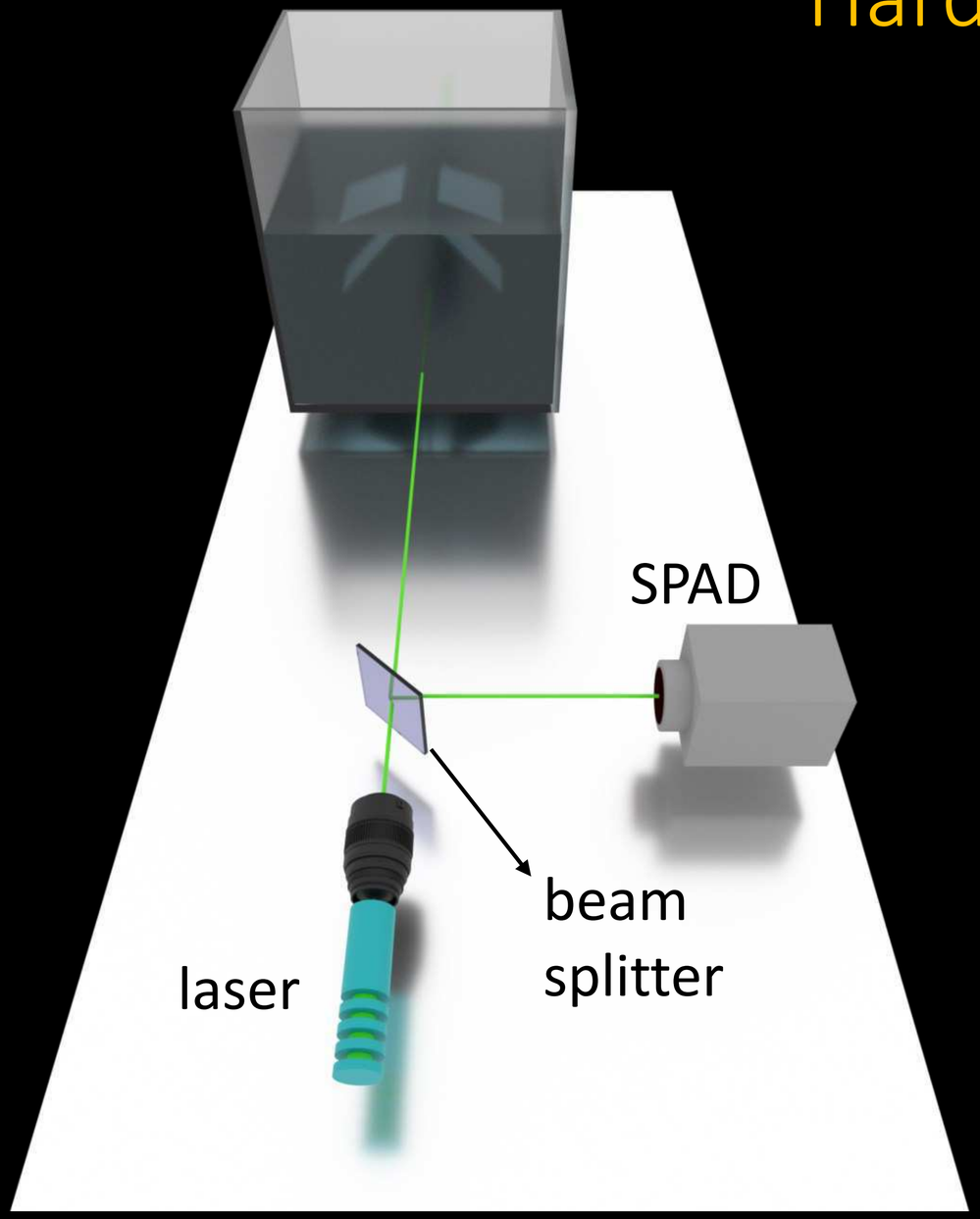
exposure= 50 ms

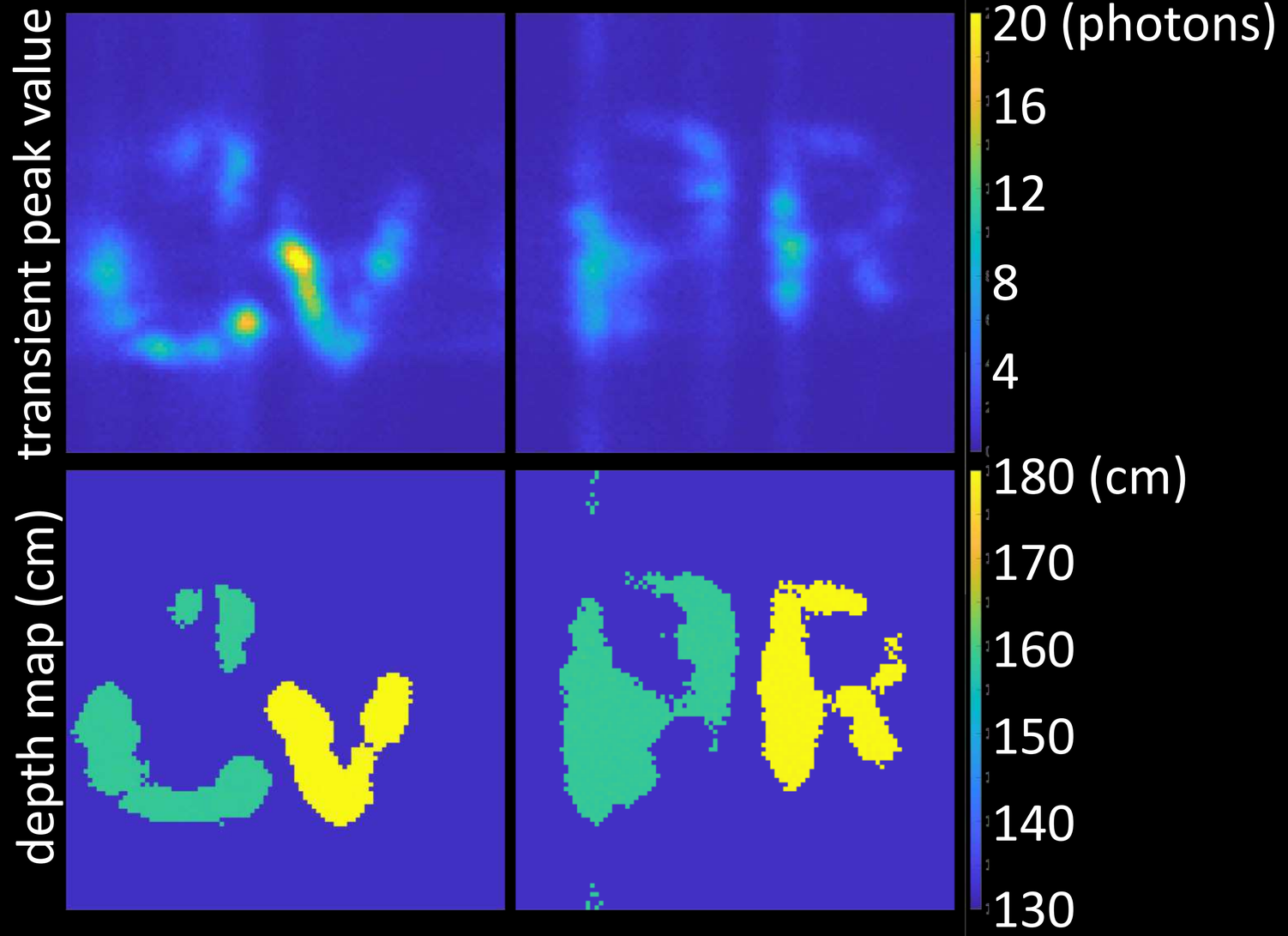


exposure= 50 ms



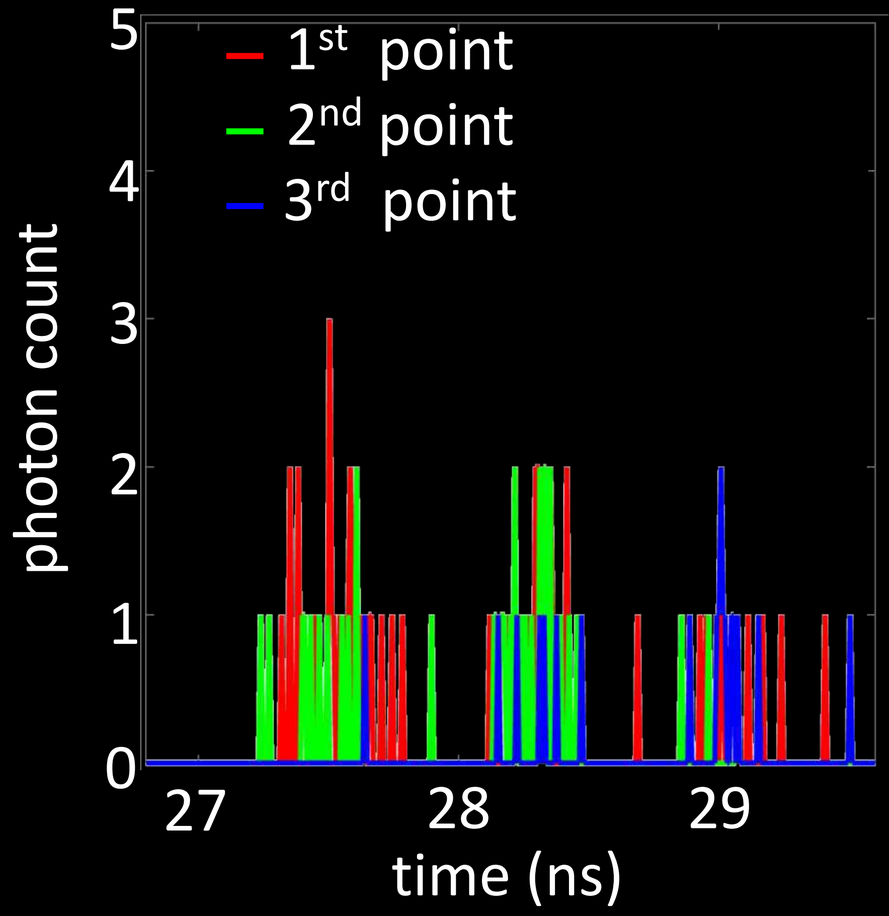
Hardware results: Lidar



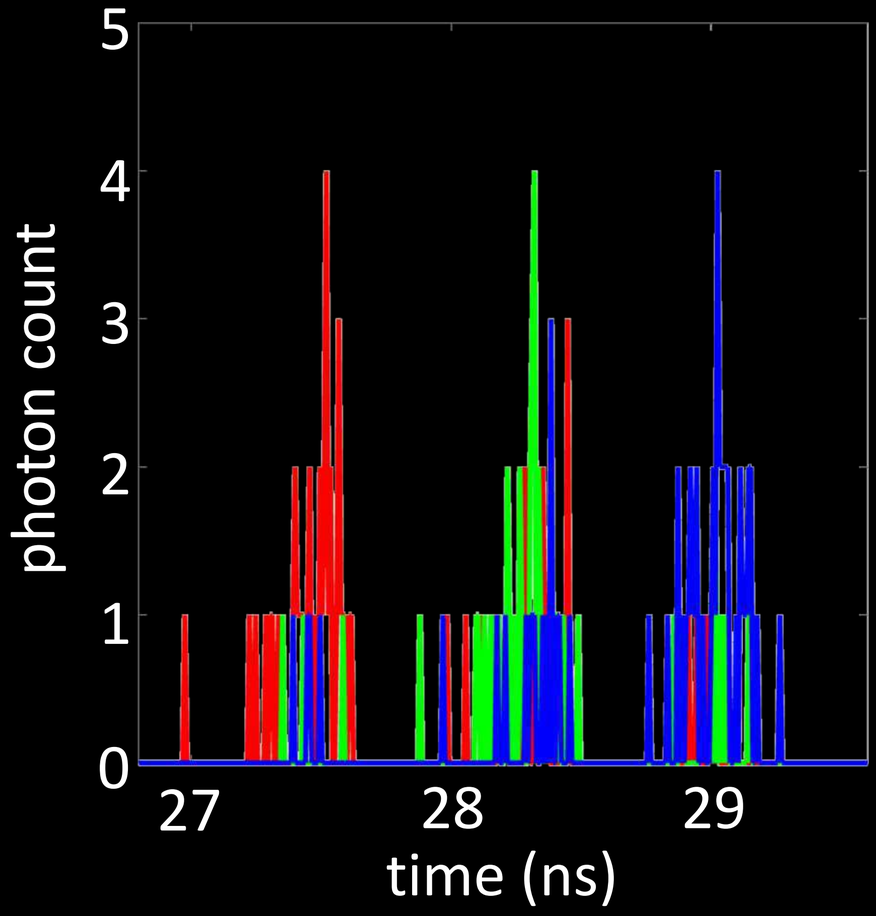


100 × faster than standard lidar

Hardware results: adaptive depth measurement



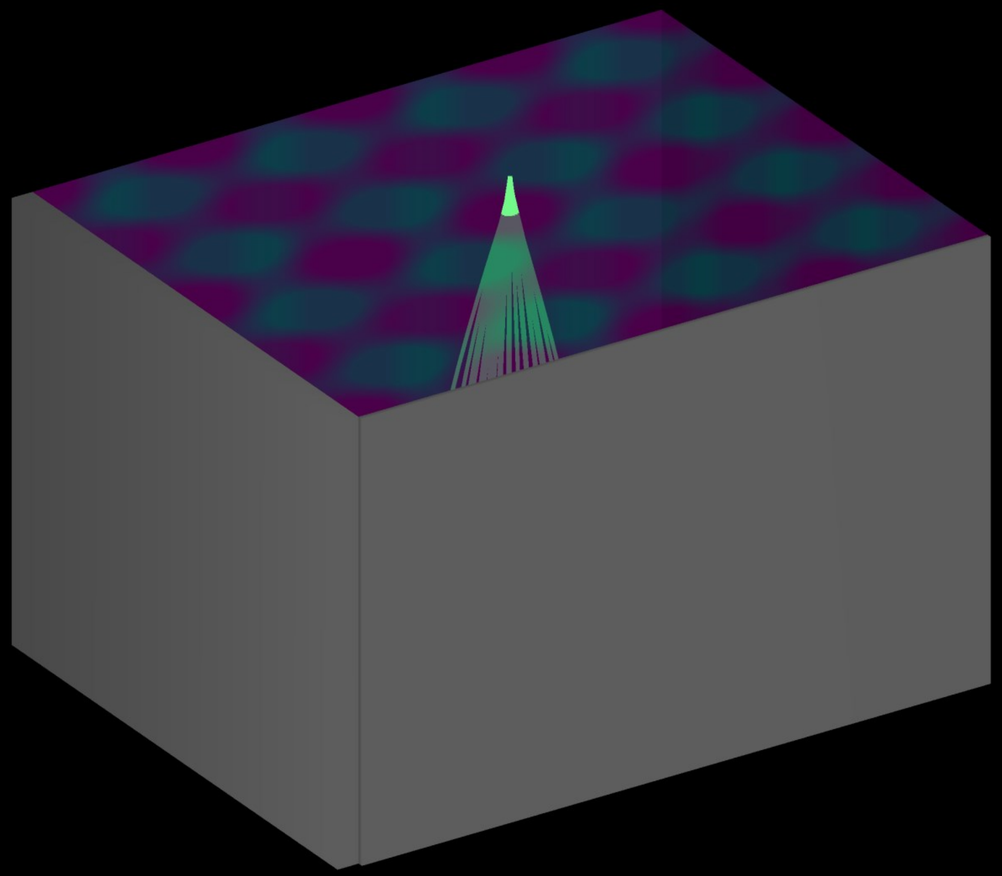
standard galvo (depth error = 51.3 cm)

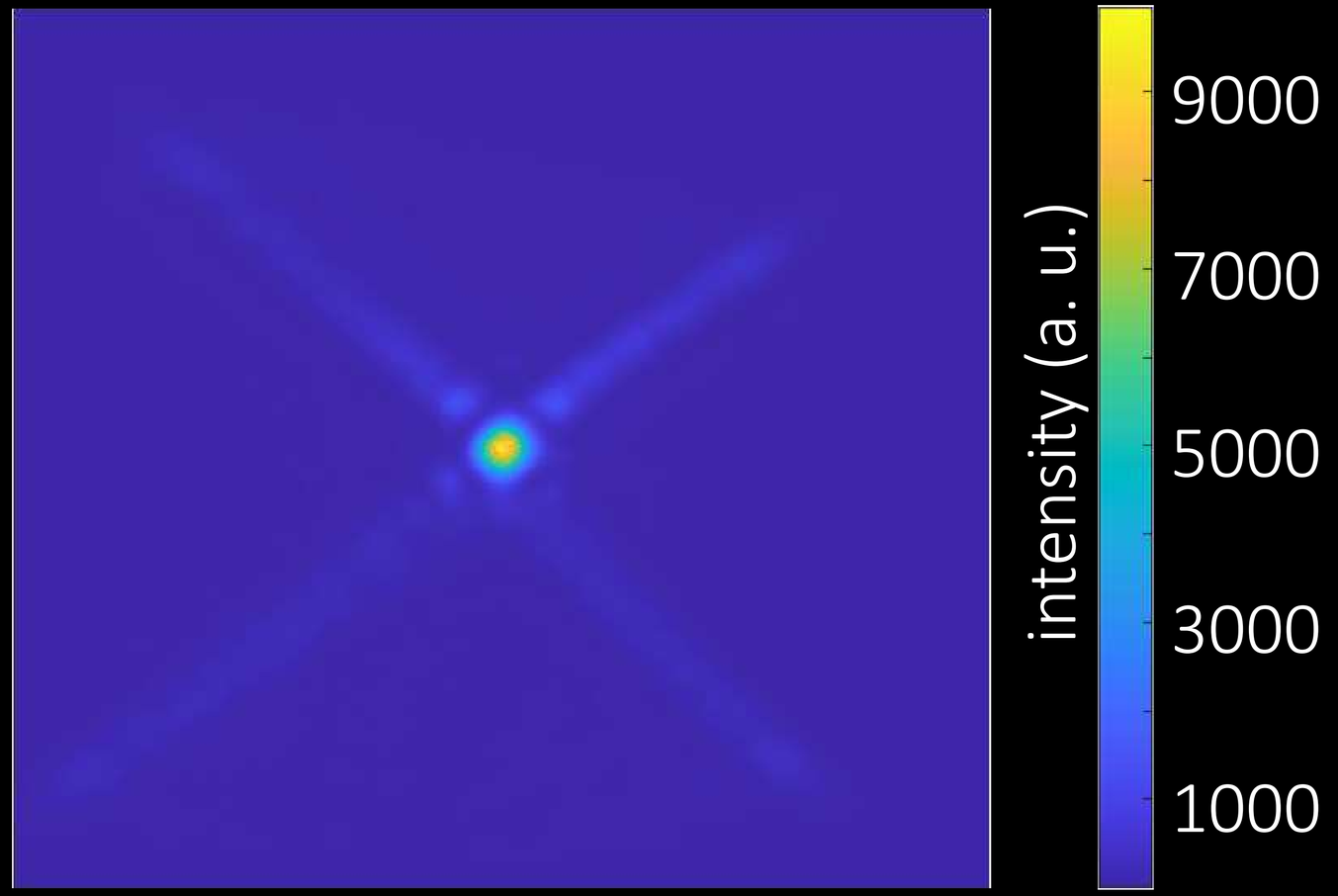


our technique (depth error = 3.28 cm)

15 × depth accuracy

Limitation

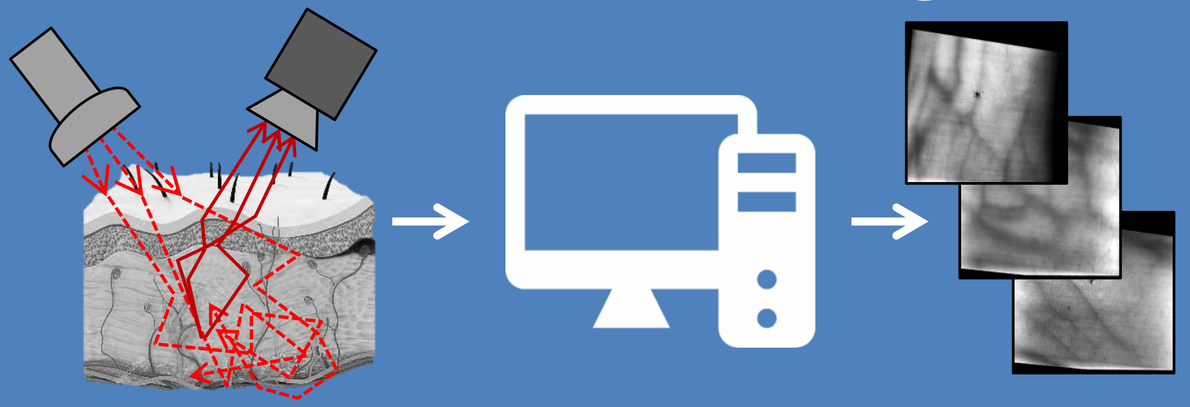




Pediredla et al. CVPR 2023.

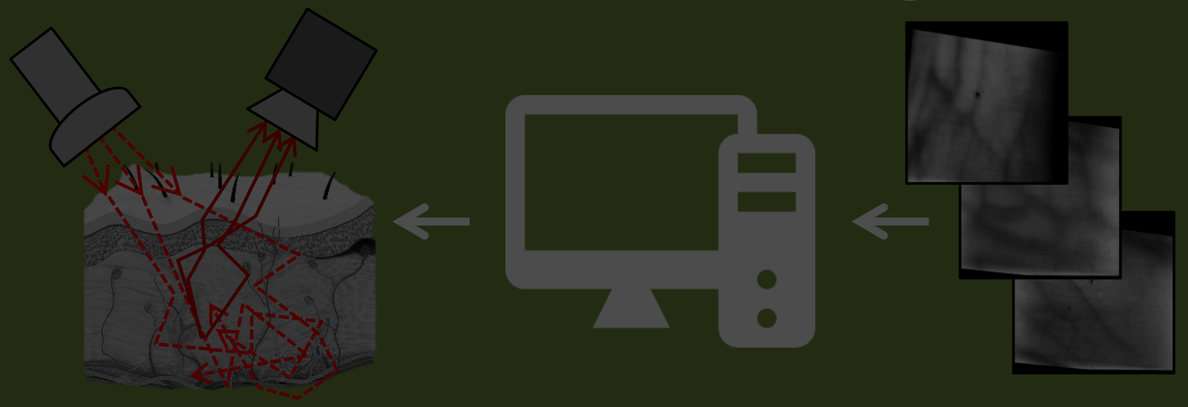
Physics-based rendering and its applications to computational imaging

forward rendering

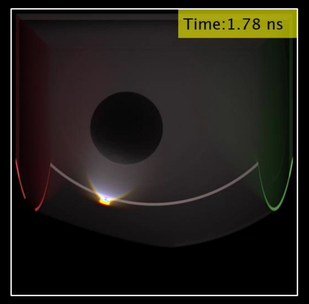


- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

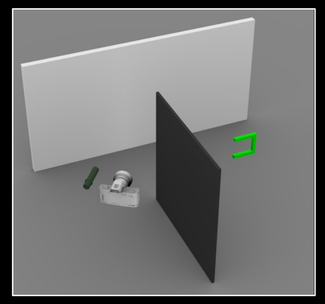
inverse rendering



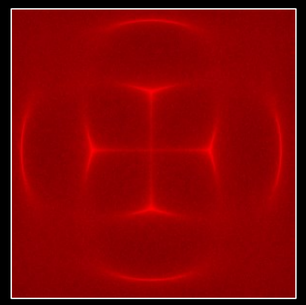
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



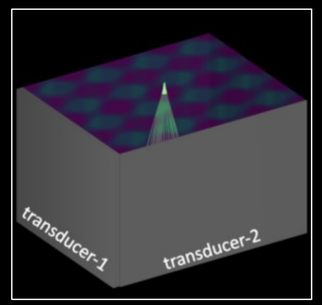
time-of-flight imaging



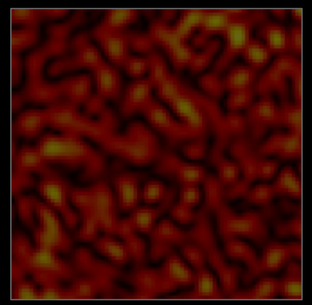
non-line-of-sight imaging



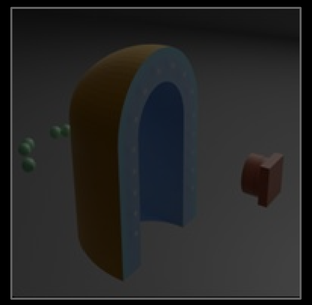
acousto-optic lensing



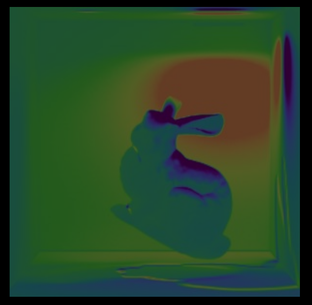
ultrafast light scanning



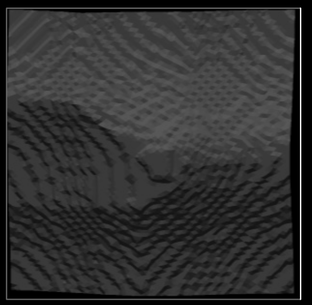
speckle imaging



tactile sensor design



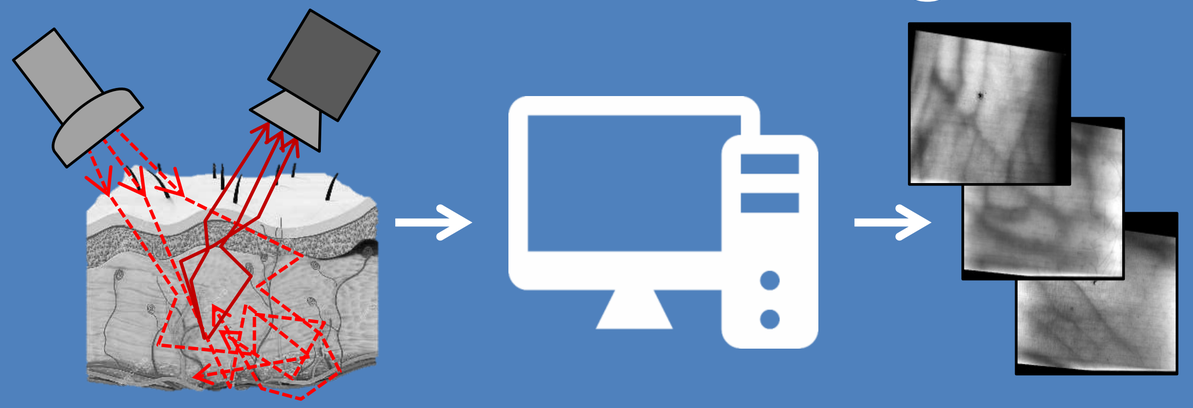
differentiable renderer



inverse problems

Physics-based rendering and its applications to computational imaging

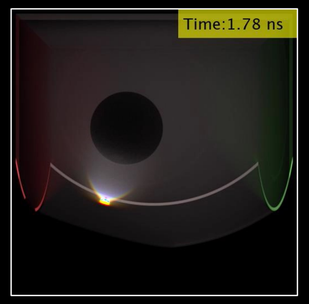
forward rendering



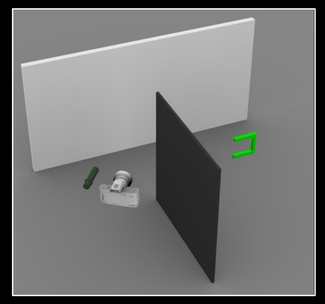
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

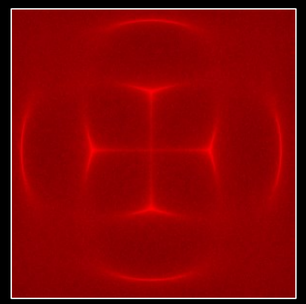
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



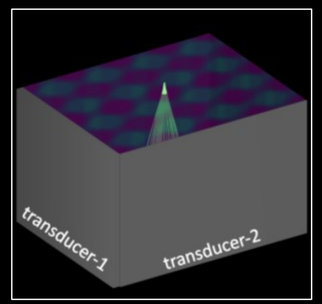
time-of-flight imaging



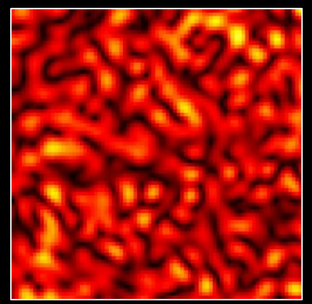
non-line-of-sight imaging



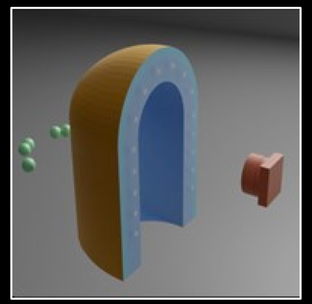
acousto-optic lensing



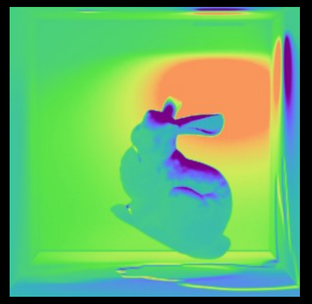
ultrafast light scanning



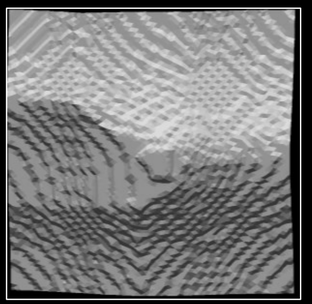
speckle imaging



tactile sensor design



differentiable renderer



inverse problems

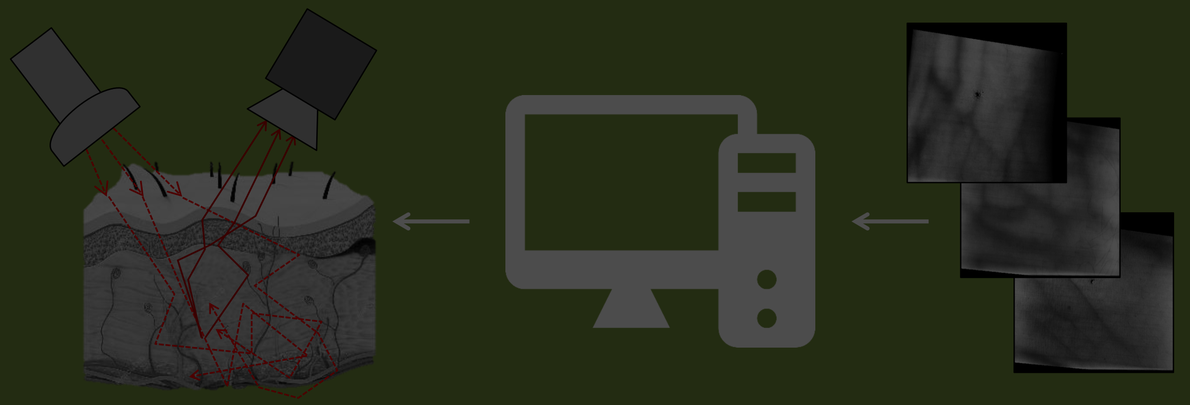
Coffee break
See you at 3.30 PM

Physics-based rendering and its applications to computational imaging

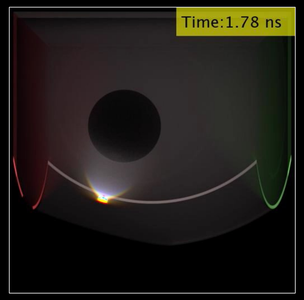
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

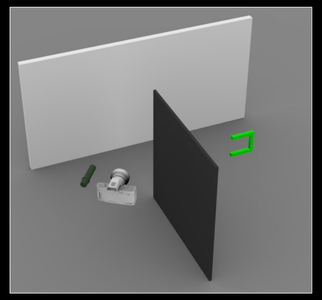
inverse rendering



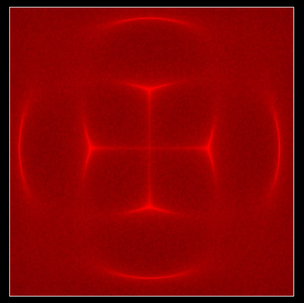
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



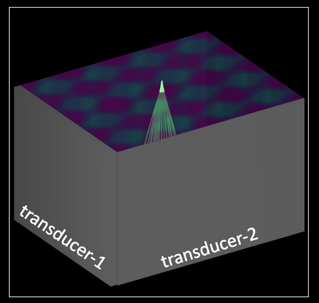
time-of-flight imaging



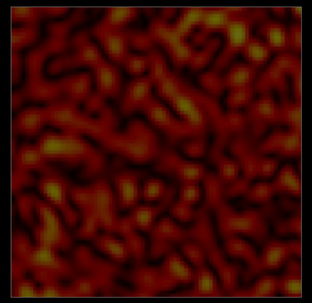
non-line-of-sight imaging



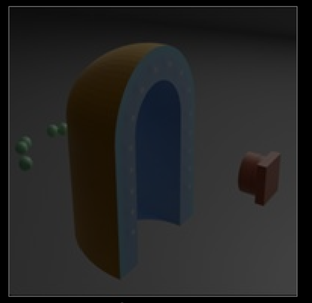
acousto-optic lensing



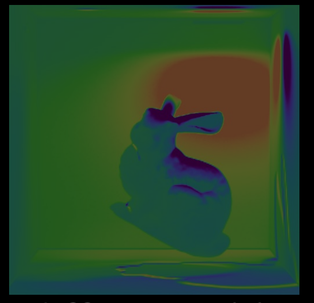
ultrafast light scanning



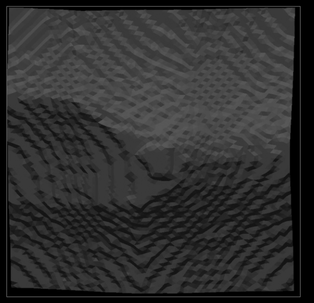
speckle imaging



tactile sensor design



differentiable renderer



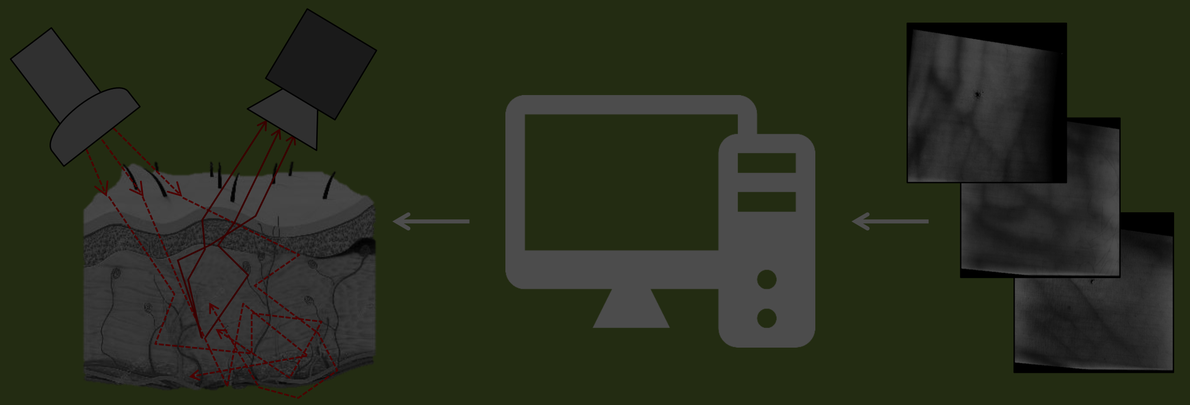
inverse problems

Physics-based rendering and its applications to computational imaging

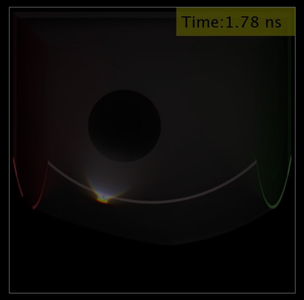
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

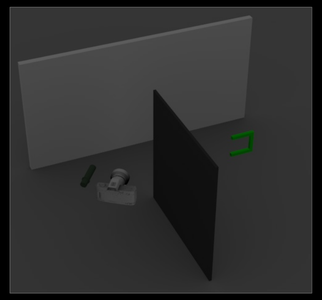
inverse rendering



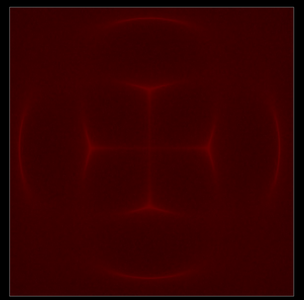
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



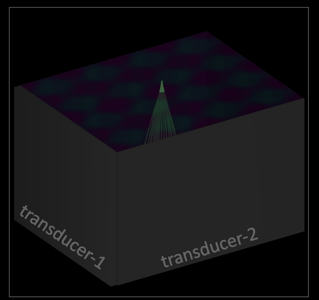
time-of-flight imaging



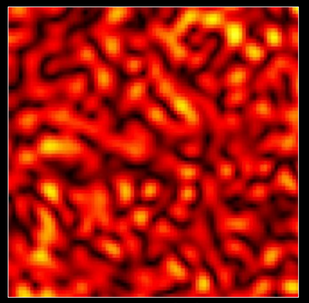
non-line-of-sight imaging



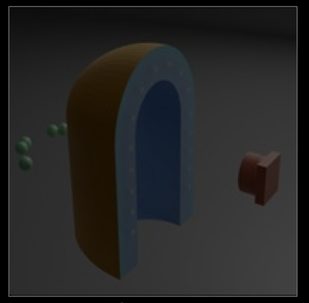
acousto-optic lensing



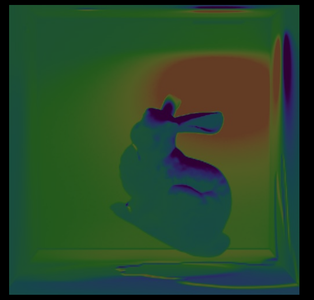
ultrafast light scanning



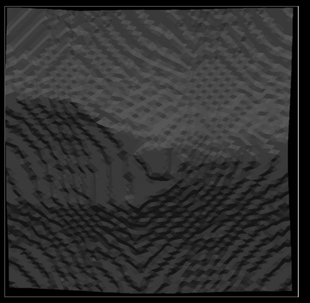
speckle imaging



tactile sensor design

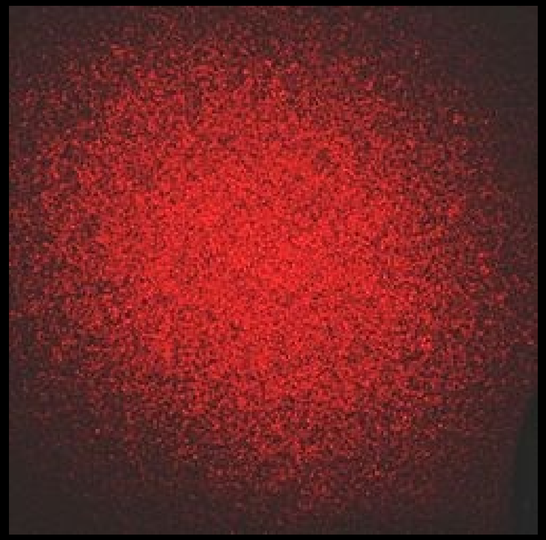


differentiable renderer



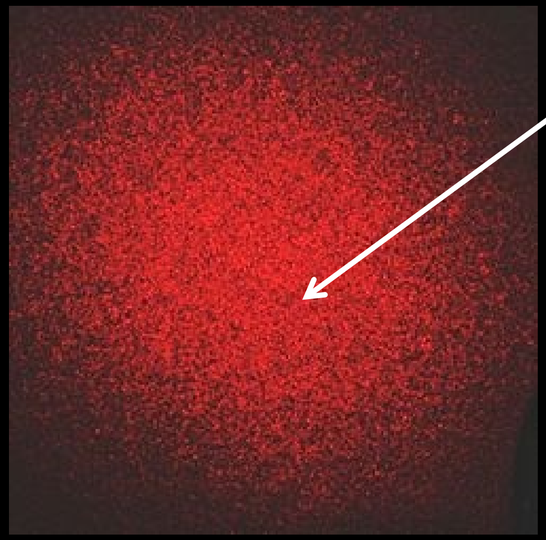
inverse problems

Rendering wave-optics effects



what real laser
images look like

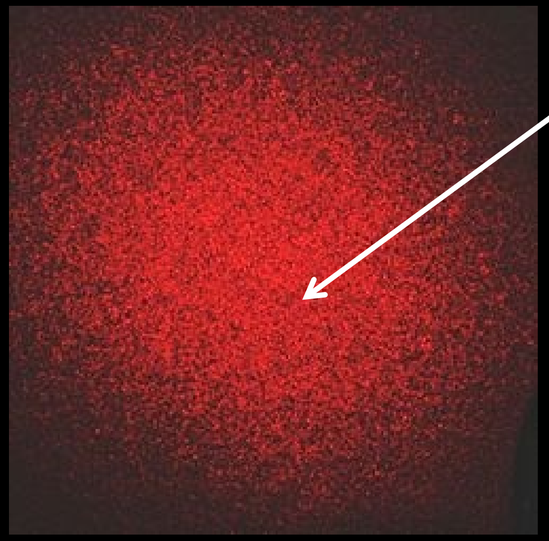
Rendering wave-optics effects



speckle: noise-like pattern

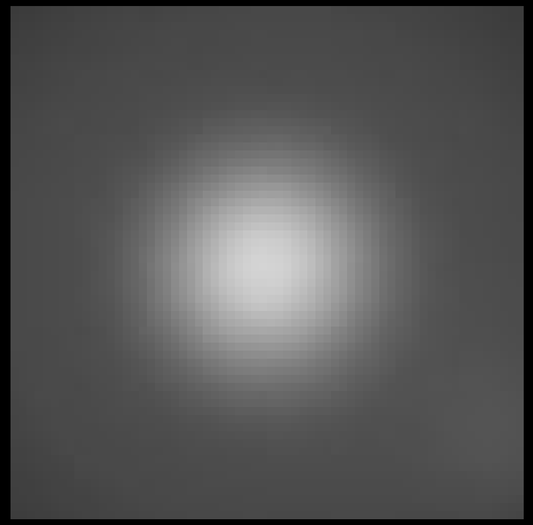
what real laser images look like

Rendering wave-optics effects



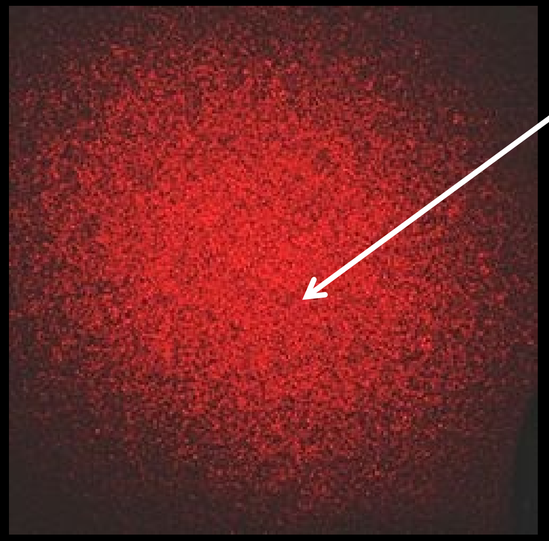
speckle: noise-like pattern

what real laser images look like



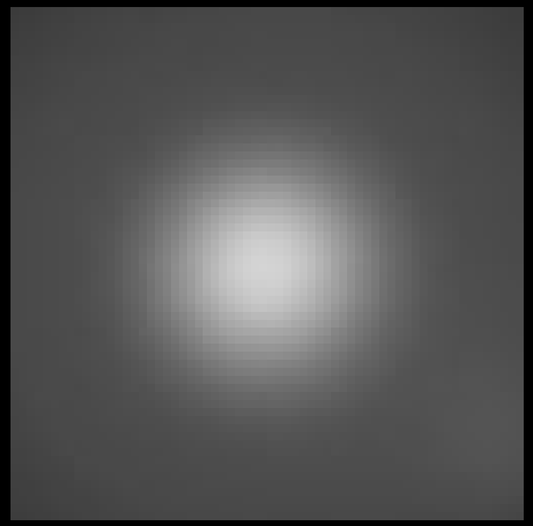
what standard Monte Carlo renderings look like

Rendering wave-optics effects

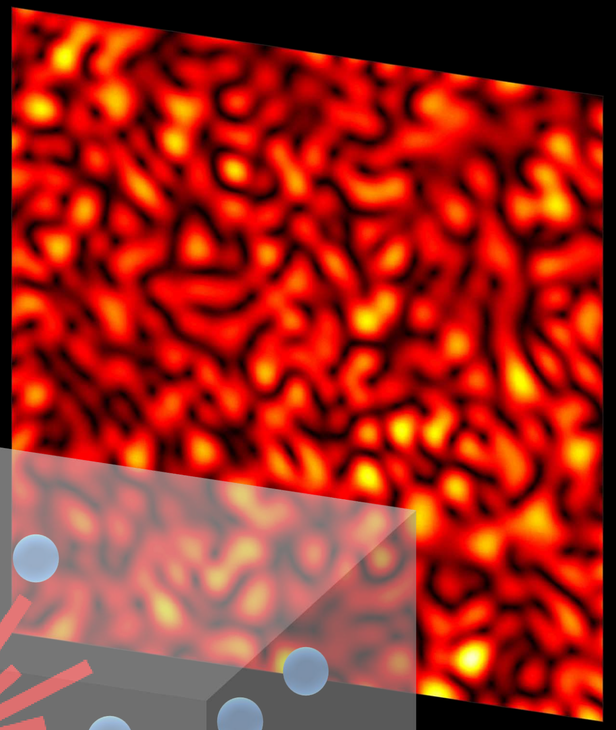


speckle: noise-like pattern

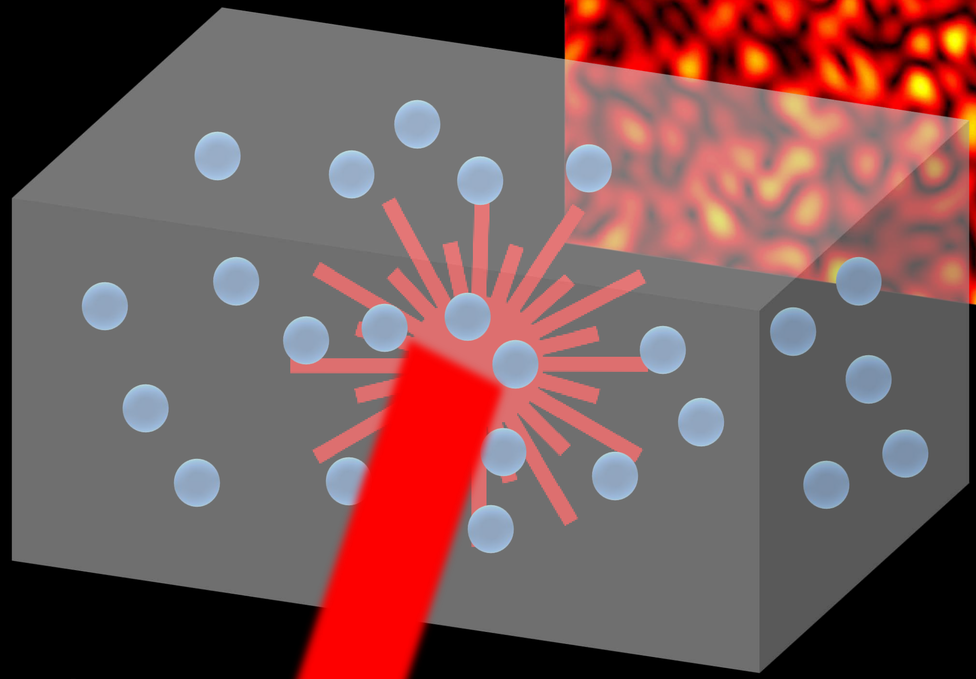
what real laser images look like



what standard Monte Carlo renderings look like



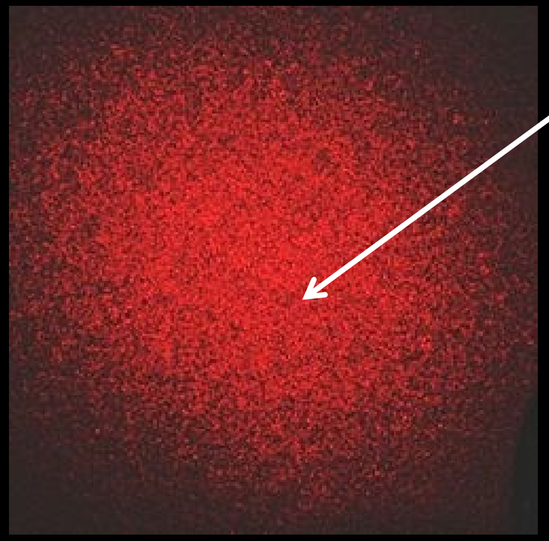
projected speckle image



laser beam

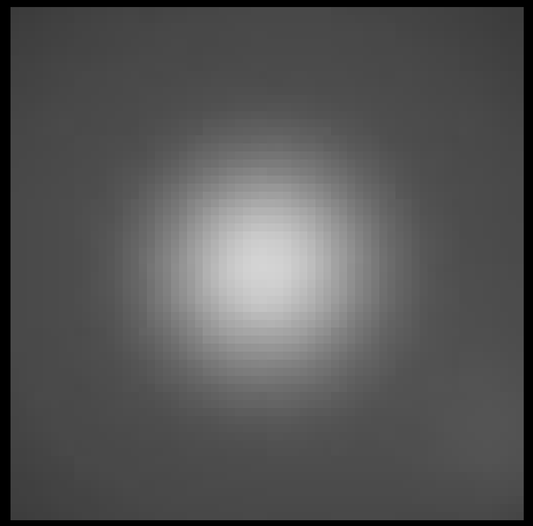
scattering volume

Rendering wave-optics effects

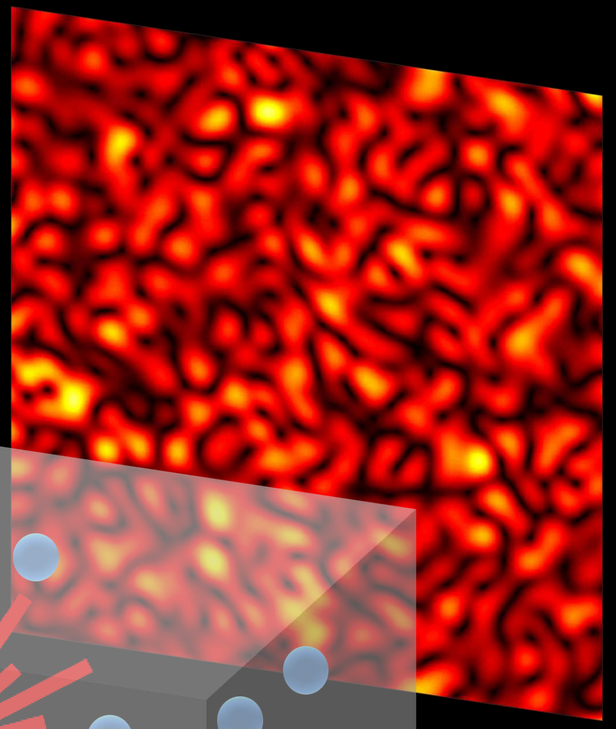


speckle: noise-like pattern

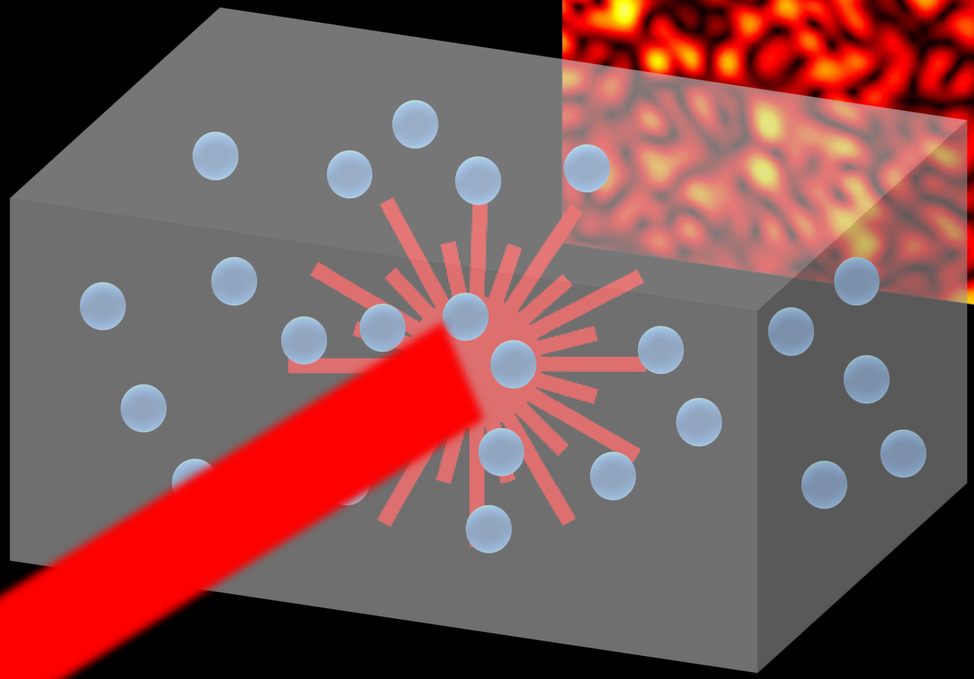
what real laser images look like



what standard Monte Carlo renderings look like



projected speckle image



scattering volume

laser beam

Applications of memory effect

SCIENTIFIC REPORTS

OPEN

Memory-effect based deconvolution microscopy for super-resolution imaging through scattering media

Eitan Edrei & Giuliano Scarcelli

Received: 26 April 2016
Accepted: 30 August 2016
Published: 16 September 2016

High-resolution imaging through turbid media is a fundamental challenge of optical sciences that has attracted a lot of attention in recent years for its wide range of potential applications. Here, we demonstrate that the resolution of imaging systems looking behind a highly scattering medium can be improved below the diffraction-limit. To achieve this, we demonstrate a novel microscopy technique enabled by the optical memory effect that uses a deconvolution image processing and thus it does not require iterative focusing, scanning or phase retrieval procedures. We show that this newly established ability of direct imaging through turbid media provides fundamental and practical advantages such as three-dimensional refocusing and unambiguous object reconstruction.

Imaging performances of traditional optical systems quickly degrade with increasing scattering[1,2], so that in the diffusive regime only low resolution optical images can be obtained[3,4]. However, recently, novel strategies such as phase conjugation, scattering-matrix inversion, ultrasonic encoding or the so-called "memory effect" have revolutionized this field showing high resolution imaging and focusing behind turbid tissue up to several scattering lengths[5–19]. In many of these studies, the ability to image through turbid media is facilitated by additional information gathered on the scattering medium. For example, placing a "guide-star" in the object plane, iterative algorithms have been used to correct for the aberrations induced by the medium[20]; or the scattered field has been recorded and reversed to focus or scan the light back onto the object plane[20,21]. Here, we use the fundamental principles of the memory effect to enable direct wide-field imaging through turbid media using deconvolution image processing and a single-shot of the scattered intensity pattern. This enabled us to achieve, through turbid

SCIENTI

OPEN

Single-sh

Xiaohan Li, Andrew Stev

Imaging through opaque sca
and astronomical imaging to
media produces scattered ligh
is contained in the signal and
studies have shown that statis
effects', allow for diffraction-li
access to) the source or scatter
be static during the measureme
imaging with coded-aperture-ba
the first time single-shot video o
important implications for a wide

Conventional optical imaging techn
approach assumes that one can measu
material intervenes, most of the light i
tor, rendering isomorphic imaging imp
imaging through opaque materials by f
incident on the scatterer[4–6], conducting
lations in the scattered light[8–11]. Of these
ing[12,13], allows for imaging through high
to the scatterer, object, or illumination. I
scatterer and/or object remain stationary
Several recent approaches have attem
stationary. For example, imposing a temp
point spread function (PSF)[13] enables fast
show that proper filtering in correlation sp
going simple (e.g. linear translational) mo
image a static object through a dynamic sca
of the scatterer as well as significant signal a
objects. Thus, none of the previously-develo
dynamic, unknown scatterer.
Here, we demonstrate a method for imag
arbitrary dynamics that may even exceed t
temporally-code the speckle image on the time
ate), we 'timestamp' the

Physica A 168 (1990) 49–65
North-Holland

LOOKING THROUGH WALLS AND AROUND CORNERS

Isaac FREUND
Department of Physics, Bar-Ilan University, Ramat-Gan, Israel

It is shown theoretically that under appropriate conditions a visually opaque scattering optical barrier can be made to serve as a thin lens which produces a real, paraxial image of objects lying behind the barrier. Preliminary experiments described which verify the validity of the underlying assumptions. The possibility made to serve as various other types of optical instruments, such as optical Fourier analyzers, theodolites, etc. Thus it is now clear that multiple should no longer be considered barriers to optical propagation, but are regarded as potential high-precision optical instruments.

1. Introduction

With the advent of radar half a century ago, detection of visually opaque barriers, such as dense cloud cover, becam randomness in size and position of water droplets which ma leads to substantial scattering of the coherent electromagne prising a radar beam. Thus, the study of coherent wa propagation through random media became an importa British Admiralty. Some of the earliest theoretical studies were carried out by Cyril Domb while seconded to the A were to form later on the basis of his very first published fields of study down to this day, and, indeed, over the last been an enormous upsurge of interest in the propagation waves in highly random media [2–53]. Here, we conside rich reservoir of new knowledge may be applied to the imaging through highly random, multiply scattering media

ARTICLES

PUBLISHED ONLINE: 29 JUNE 2015 | DOI: 10.1038/NPHYS3373

Translation correlations in anisotropica scattering media

Benjamin Judkewitz[1,2]*†, Roarke Horstmeyer[2†], Ivo M. Vellekoop[3], Ioannis N. Pap and Changhuei Yang[2]

Controlling light propagation across scattering media by wavefront shaping holds great prom communications and imaging applications. But, finding the right shape for the wavefront is a chal between input and output scattered wavefronts (that is, the transmission matrix) is not known. Cor matrices, especially the so-called memory effect, have been exploited to address this limitation. memory effect applies to thin scattering layers at a distance from the target, which preclud scattering media, such as fog and biological tissue. Here, we theoretically predict and experimentally matrix correlations within thick anisotropically scattering media, with important implications for adaptive optics.

Focusing light through strongly scattering media is an important goal in optical imaging and communication. Long considered impossible, recent advances in the field of wavefront shaping[1,2] changed this view by demonstrating that diffuse light can be focused through inhomogeneous media—as long as the correct input wavefront is used. With direct optical access to the target plane, the correct wavefront can be obtained by iterative optimization[3], phase conjugation[4], or by measuring the transmission matrix[4,5]. In many imaging scenarios, however, there is no direct access to the target plane. In those cases, nonlinear[6], fluorescent[7], kinematic[8,9], acousto-optic[10–12] and photo-acoustic[13–15] guide stars can be used as reference beacons. However, these techniques provide wavefront information for only one target location at a time. Although transmission matrices can be sampled quickly with a photo-acoustic approach[16], this method requires absorbing samples. As a result, many samples' transmission matrices can be sampled only sparsely. Correlations within a transmission matrix can compensate for sparse sampling and could enable high-speed imaging. One of the most widely known transmission matrix correlations is the so-called 'memory effect'[17,18], which describes the following phenomenon: when an input wavefront reaching a diffusing sample is tilted within a certain angular range, the output wavefront is equally tilted, resulting in the translation of the far-field speckle pattern at a distance behind the sample (see Fig. 1).

The translation distance within which this effect holds (that target area of interest, the correlations pre memory effect should be minimal[27]. Here whether there are other correlations that We show that significant transmission c thick scattering media at zero distance is directional.

Traditional memory effect

Although the memory effect has been exten principles[18], these derivations relied on as scattering—which does not apply to many we will approach the problem without r instead using transmission matrices. Spec in the matrix $T_t = T(x_t, x_b)$, which defines the spatial input and output optical mo (that is, from input plane A to output of graphical representation, we assum dimensional (1D) wavefronts in a 2D geo will be generalizable to 2D wavefronts in its discrete nature, the transmission mat to experimental observation. We will f analyse the traditional memory effect, an correlations in thick anisotropic media.

When assuming highly randomizing plete measurement of input and output

ARTICLES

PUBLISHED ONLINE: 31 AUGUST 2014 | DOI: 10.1038/NPHOTON.2014.189

nature photonics

Non-invasive single-shot imaging through scattering layers and around corners via speckle correlations

Ori Katz[1,2]*, Pierre Heidmann[1], Mathias Fink[1] and Sylvain Gigan[1,2]

Optical imaging through and inside complex samples is a difficult challenge with important applications in many fields. The fundamental problem is that inhomogeneous samples such as biological tissue randomly scatter and diffuse light, preventing the formation of diffraction-limited images. Despite many recent advances, no current method can perform non-invasive imaging in real-time using diffused light. Here, we show that, owing to the 'memory-effect' for speckle correlations, a single high-resolution image of the scattered light, captured with a standard camera, encodes sufficient information to image through visually opaque layers and around corners with diffraction-limited resolution. We experimentally demonstrate single-shot imaging through scattering media and around corners using spatially incoherent light and various samples, from white paint to dynamic biological samples. Our single-shot lensless technique is simple, does not require wavefront-shaping nor time-gated or interferometric detection, and is realized here using a camera-phone. It has the potential to enable imaging in currently inaccessible scenarios.

Diffraction-limited optical imaging is an indispensable tool in many fields of research. Unfortunately, however, the inherent inhomogeneity of complex samples such as biological tissues induces light scattering, which diffuses any optical beam into a complex speckle pattern[1], limiting the resolution and penetration depth of optical imaging techniques[2]. Many approaches to overcome this fundamental, yet practical, problem have been put forward over the years, with pioneering experiments in holography dating back to just a few years after the invention of the laser[3,4]. However, to date, no approach allows real-time non-invasive imaging using diffused light. Modern techniques that are based on using only unscattered, 'ballistic' light, such as optical coherence tomography and two-photon microscopy, have proven very useful, but are inherently limited to shallow depths where a measurable amount of unscattered photons is present. Adaptive optics techniques[5] can near-perfectly correct low-order aberrations using deformable mirrors, but require the presence of a bright point-source 'guide star' or a high initial image contrast[6]. Recent exciting advances in controlled wavefront shaping[7] have allowed focusing and imaging through highly scattering samples[8–26]. However, these techniques either require initial access to both sides of the scattering medium[8–15], the presence of a guide-star or a known object[16–19], or a long acquisition sequence that involves the projection of a large number of optical patterns[20–26]. A recent breakthrough approach reported by Bertolotti *et al.* has removed the requirement for a guide-star or a

of the scattered light that diffuses through a scattering medium encodes sufficient information to image through it, with diffraction-limited resolution. Specifically, derived from concepts used in 'stellar speckle interferometry'[29–31] and the angular 'memory-effect' for speckle correlations[32–34] exploited in Bertolotti's technique[27], we show that the autocorrelation of the scattered light pattern (Fig. 1c) is essentially identical to the autocorrelation of the object's image itself (Fig. 1g). We then reconstruct the object's image from its autocorrelation using an iterative Fienup-type algorithm[35]. As proofs of concept, we experimentally demonstrate our single-shot non-invasive technique through a variety of highly scattering samples, from optical diffusers to dynamically varying biological tissues. In addition, we demonstrate imaging 'around corners' by recording the diffuse light back-scattered off white-painted 'walls'.

Principle

A schematic of the experiment for imaging through a scattering medium, as well as a numerical example, are presented in Fig. 1a–d. An object is hidden at a distance u behind a highly scattering medium of thickness L. The object is illuminated by a spatially incoherent, narrowband source, and a high-resolution camera that is placed at a distance v on the other side of the medium records the pattern of the scattered light that has diffused through the scattering medium. Although the raw recorded camera image is a low-contrast, random and seemingly information-less image (Fig. 1b), its autocorrelation (Fig. 1c) is essentially identical to the object's autocorrelation (Fig. 1g) that would have been imaged by an aberration-free diffraction-

LETTER

doi:10.1038/nature11578

-invasive imaging through opaque scattering s

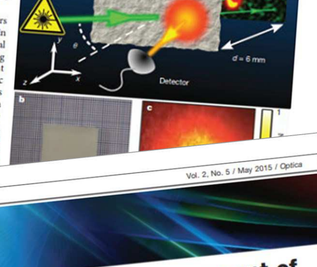
ottil[1,2]*, Elbert G. van Putten[1]†*, Christian Blum[3], Ad Lagendijk[1,4], Willem L. Vos[1] & Allard P. Mosk[1]

tical imaging techniques, such as optical coher-[1–3], are essential diagnostic tools in many disci-life sciences to nanotechnology. However, present ble to image through opaque layers that scatter ht[4,5]. Even a very thin layer of a scattering ma-paque and hide any objects behind it*. Although been made recently with methods such as ghost front shaping[6–11], present procedures are still y require either a detector[11] or a nonlinear behind the scattering layer. Here we report at allows non-invasive imaging of a fluore-mpletely hidden behind an opaque scattering the object with laser light that has passed layer. We scan the angle of incidence of xt the total fluorescence of the object from fected signal, we obtain the image of the iterative algorithm[14,15]. As a proof of con-ed image of a fluorescent object, compar-etres) to a typical human cell, hidden opaque optical diffuser, and an image sample enclosed between two opaque non-invasive imaging through strongly neralized to other contrast mechanisms

day, scattering of light severely impairs attering medium allows light to pass in t completely scrambles all the spatial as proved very successful in imaging to separate the small amount of light wing to random scattering (ballistic und using a gated technique such as In this way it is possible to obtain parent media, but for stronger scat-and prevents present ring medium can be oes not allow one to

light is scrambled by scattering and through the diffuser we cannot measure any spatial information about the shape of the object (Fig. 1c). However, the total amount of transmitted fluorescence can be measured and retains the information on the overlap between the object's fluorescent response, $O(r)$, and the speckle intensity, $S(r)$, where r is the vector of spatial coordinates.

In our measurement procedure, we scan the angle of incidence, $\theta = (\theta_x, \theta_y)$, of the laser beam using a pair of scanning galvanic mirrors (Supplementary Information). Although the speckle illuminating the object might appear random, it contains correlations that can be exploited. In particular, the angular correlation known as the memory effect[12,22] means that rotating the incident beam over small angles θ does not change the resulting speckle pattern but only translates it over a distance $\Delta r \approx 0d$. Therefore, up to a proportionality constant that we will set to 1, the total measured fluorescence as a function of the incident angle is given by



Vol. 2, No. 5 / May 2015 / Optica 424

ica

correlation resolution enhancement of d fluorescence imaging

ELBERT G. VAN PUTTEN,[1,2] JACOPO BERTOLOTTI,[1,3] AD LAGENDIJK,[1] AND ALLARD P. MOSK[1]
ms (COPS), MESA+ Institute for Nanotechnology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
s Research Laboratories, 5656 AE Eindhoven, The Netherlands
ics and Astronomy Department, University of Exeter, Stocker Road, Exeter EX4 4QL, UK
yilmaz@utwente.nl

14; revised 20 March 2015; accepted 21 March 2015 (Doc. ID 226377); published 27 April 2015

fluorescence imaging is essential in nanoscience and biological sciences. Due to the diffraction limit, aging systems can only resolve structures larger than 200 nm. Here, we introduce a new fluorescence d that enhances the resolution by using a high-index scattering medium as an imaging lens. we achieve a wide field of view. We develop a new image reconstruction algorithm that converges e object structures. We collect two-dimensional fluorescence images of a collection of 100 nm diameter spheres, and demonstrate a deconvolved Abbe resolution of 116 nm with a field of view of ithm is robust against optical aberrations and stage drifts, and therefore is well suited to Our method is robust against optical aberrations and stage drifts, and therefore is well suited to ctures with high resolution under ambient conditions.

0.0180) Microscopy; (290.0290) Scattering; (180.2520) Fluorescence microscopy; (110.6150) Speckle imaging; solution; (100.5070) Phase retrieval.

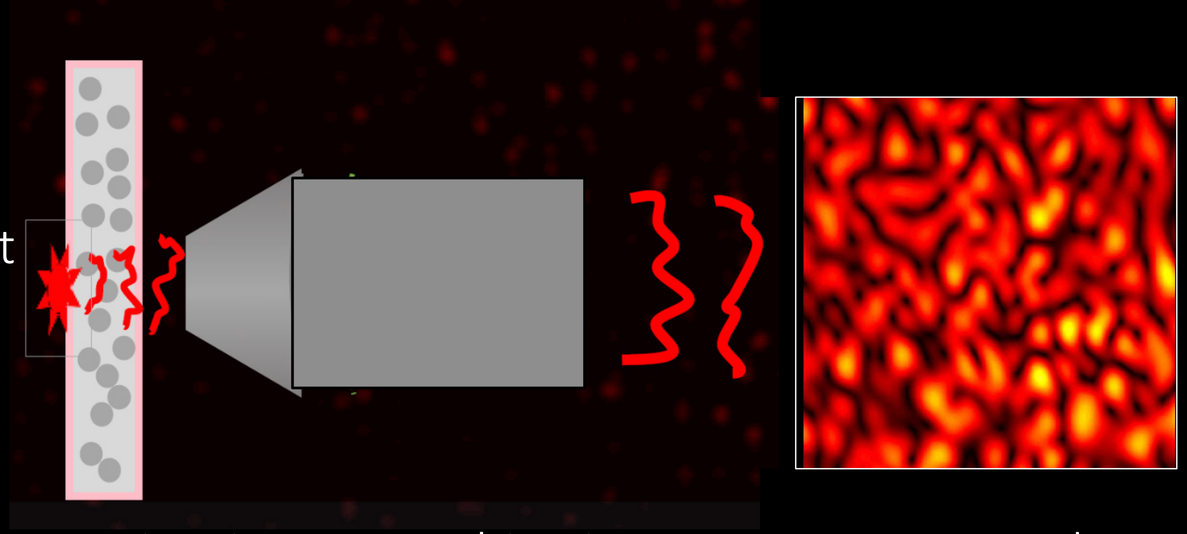
0.1364/OPTICA.2.000424

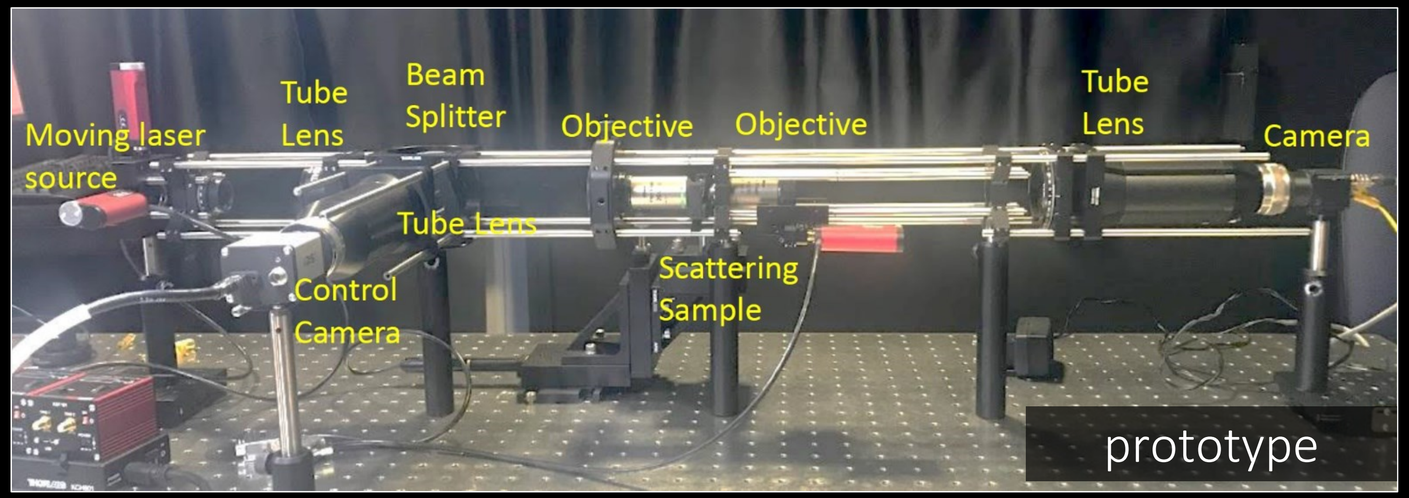
microscope produces images with a re-e numerical aperture (NA) of the imaging maging lens is defined by the highest wave in the transversal direction. Many meth-ed that enable optical resolution beyond a conventional optical microscope. These escent waves with near-field scanning op-M) [2]; moiré fringes such as in structured y (SIM) [3,4] or blind structured illumi-d-SIM) [5]; nonlinear optical phenomena M [6] or stimulation emission depletion 8]; and specific photophysical properties astic optical reconstruction microscopy ctivated localization microscopy (PALM) on optical fluctuation imaging (SOFI) DM has a field of view limited by the scan greater than a conventional optical micro-Moreover, a scanning probe usually greatly nt itself. SIM provides a resolution that es a precise knowledge of the illuminating

samples with a low damage threshold. STED, STORM, PA and SOFI require dyes with specific photophysical prope Although the aforementioned far-field microscopy me realize an optical resolution beyond the diffraction limit, t olution remains strongly dependent on the NA of their co tional optics.

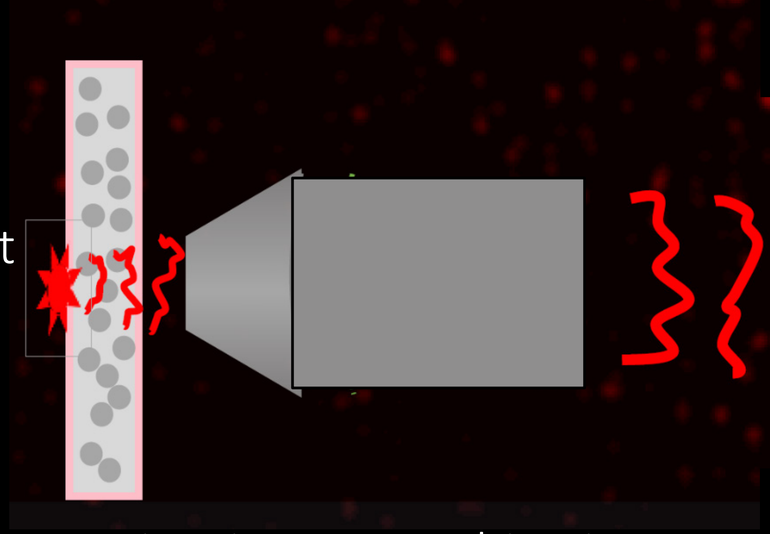
It has been shown that a scattering medium enhances resolution for acoustic waves [12], microwaves [13], a waves [14,15], and turns a high-index substrate into NA solid immersion lens by breaking the translational in on the interface of the substrate [16]. Coherent light illu on such a scattering medium generates a speckle pattern rently randomly distributed bright and dark regions b scattering medium. Correlations of speckle patterns ar obtain images of astronomical objects through turbule sphere with diffraction-limited resolution [17,18]. The exploiting correlations between such speckle patterns images of microscopic objects has started a new clas microscopy [19]. Within a speckle pattern there is a effect called the optical memory effect [20,21] that h been exploited for optical imaging using a high-NA sc [16] and through scattering media [22–28]. A tilt of beam within an angle of $\Delta\theta < \lambda/2\pi d$ (λ is the wave

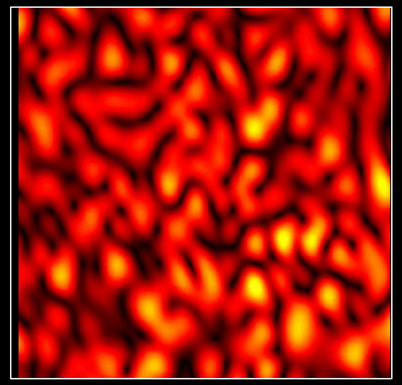
Speckle-based fluorescence microscopy





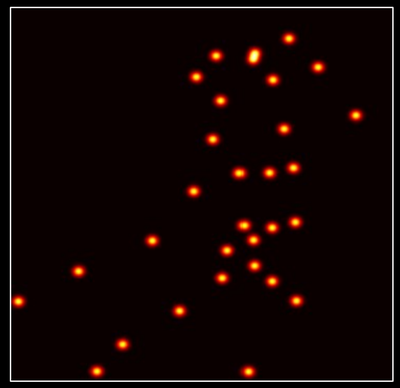
Speckle-based fluorescence microscopy



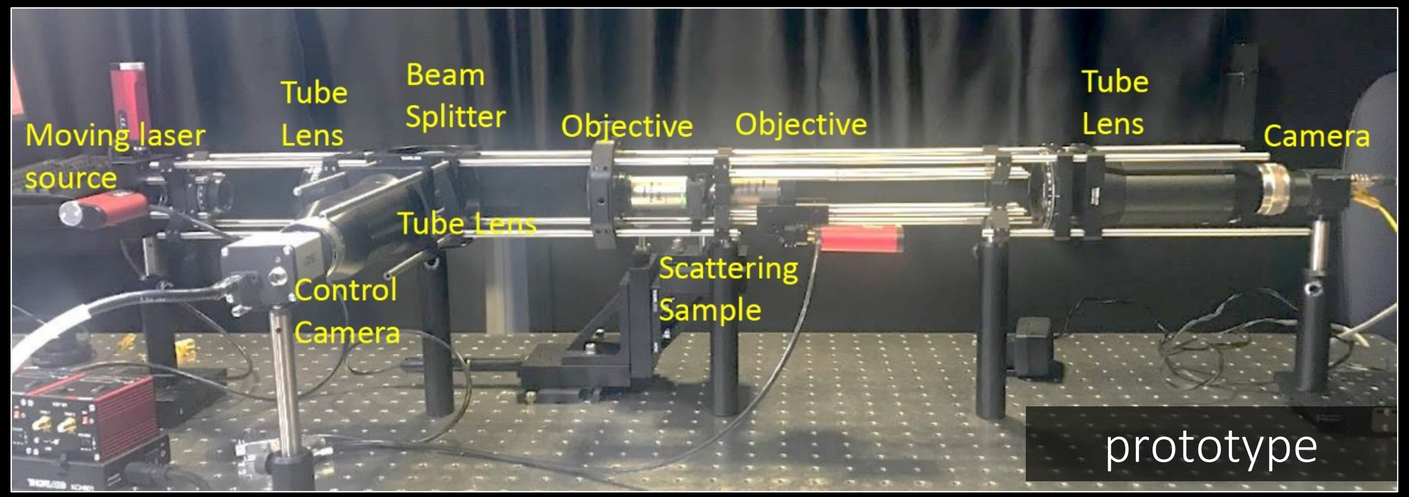


captured image

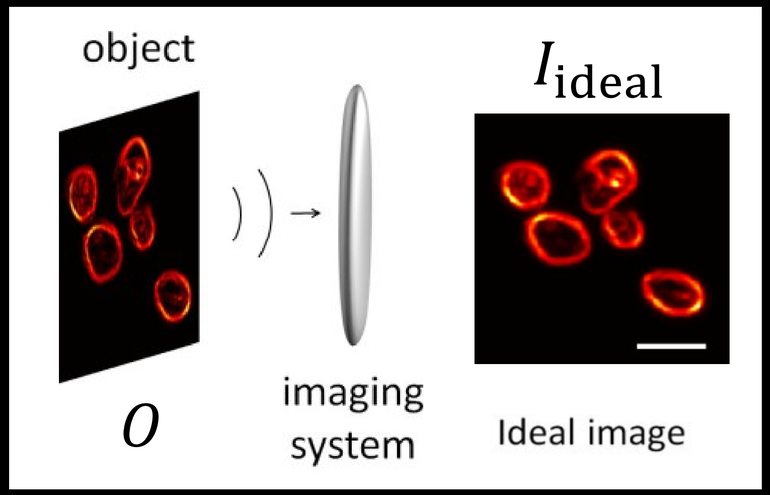
→ ? →

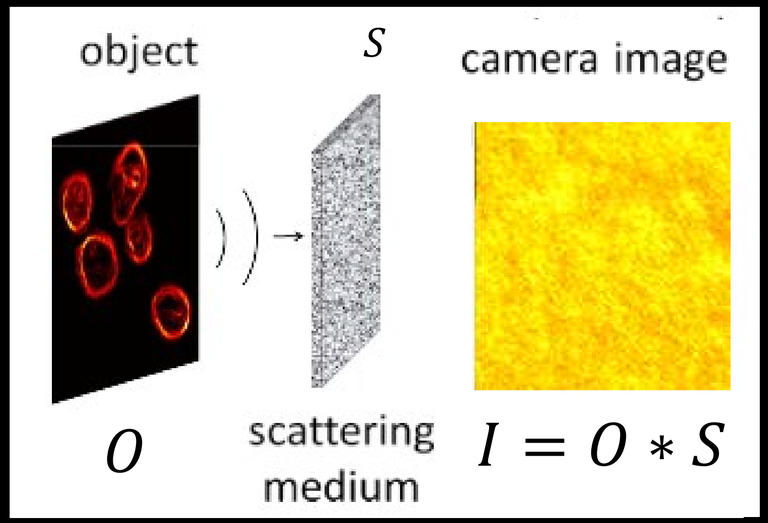


scattering-free image



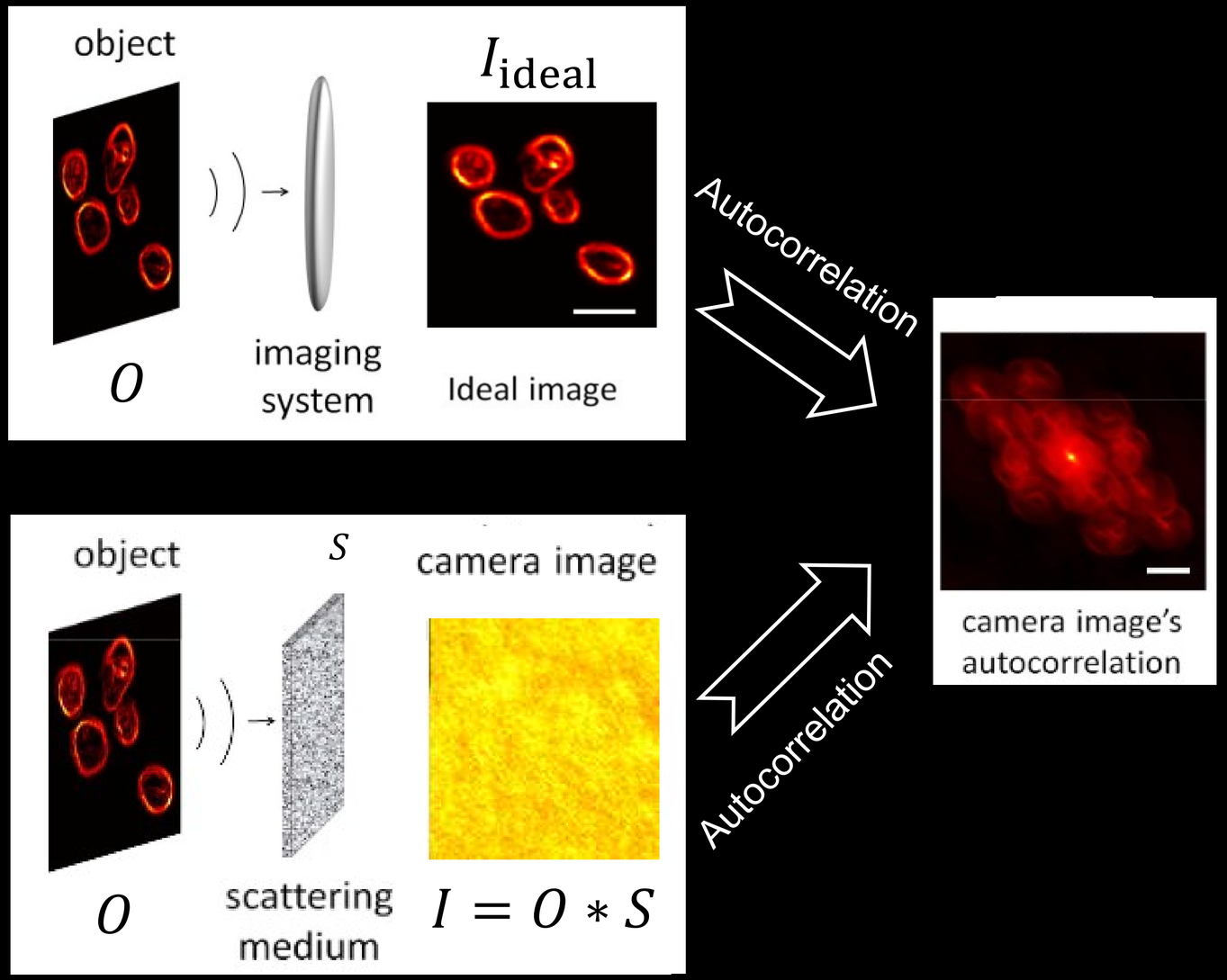
Use the memory effect to image through scattering





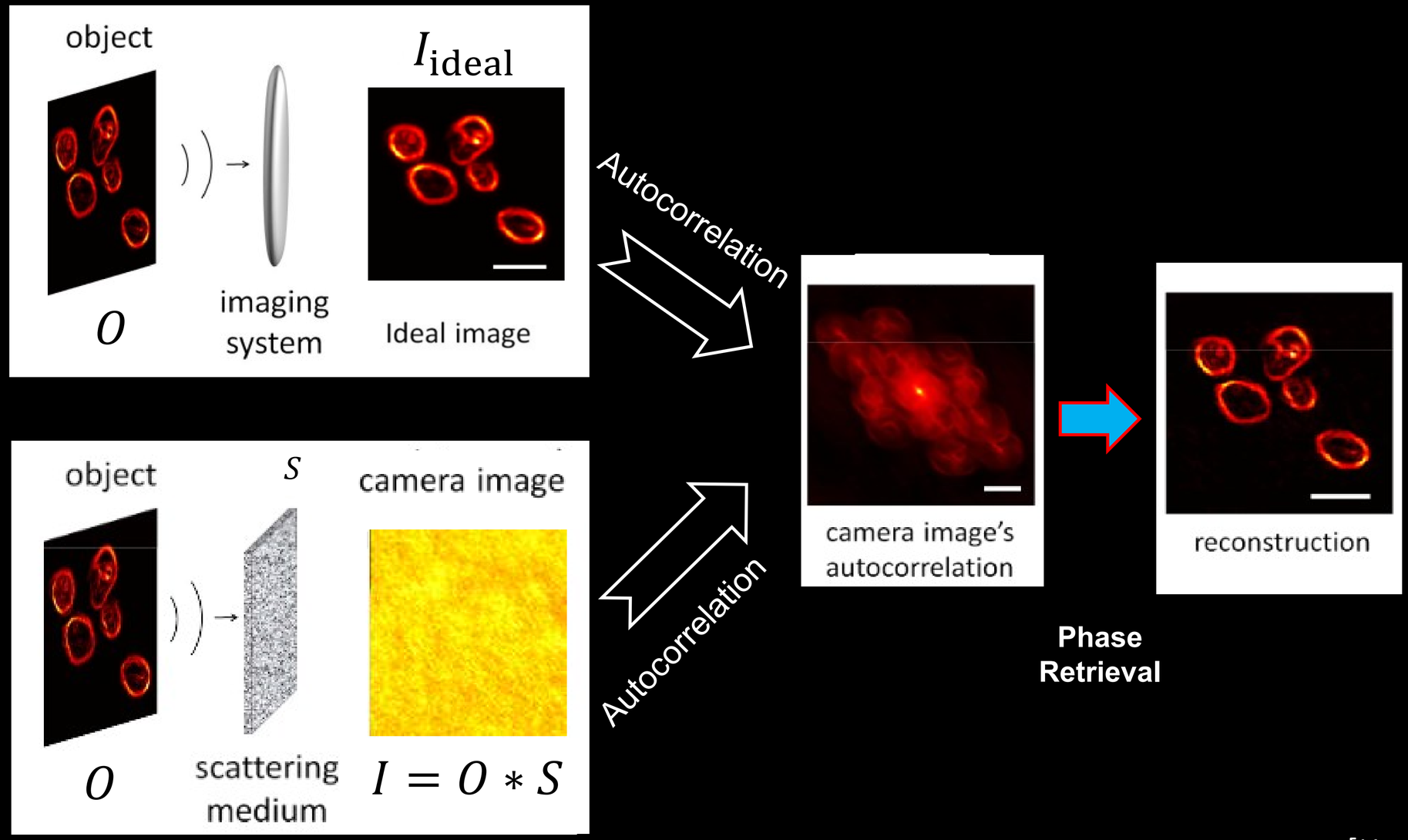
[Katz et al., 2014]

Use the memory effect to image through scattering



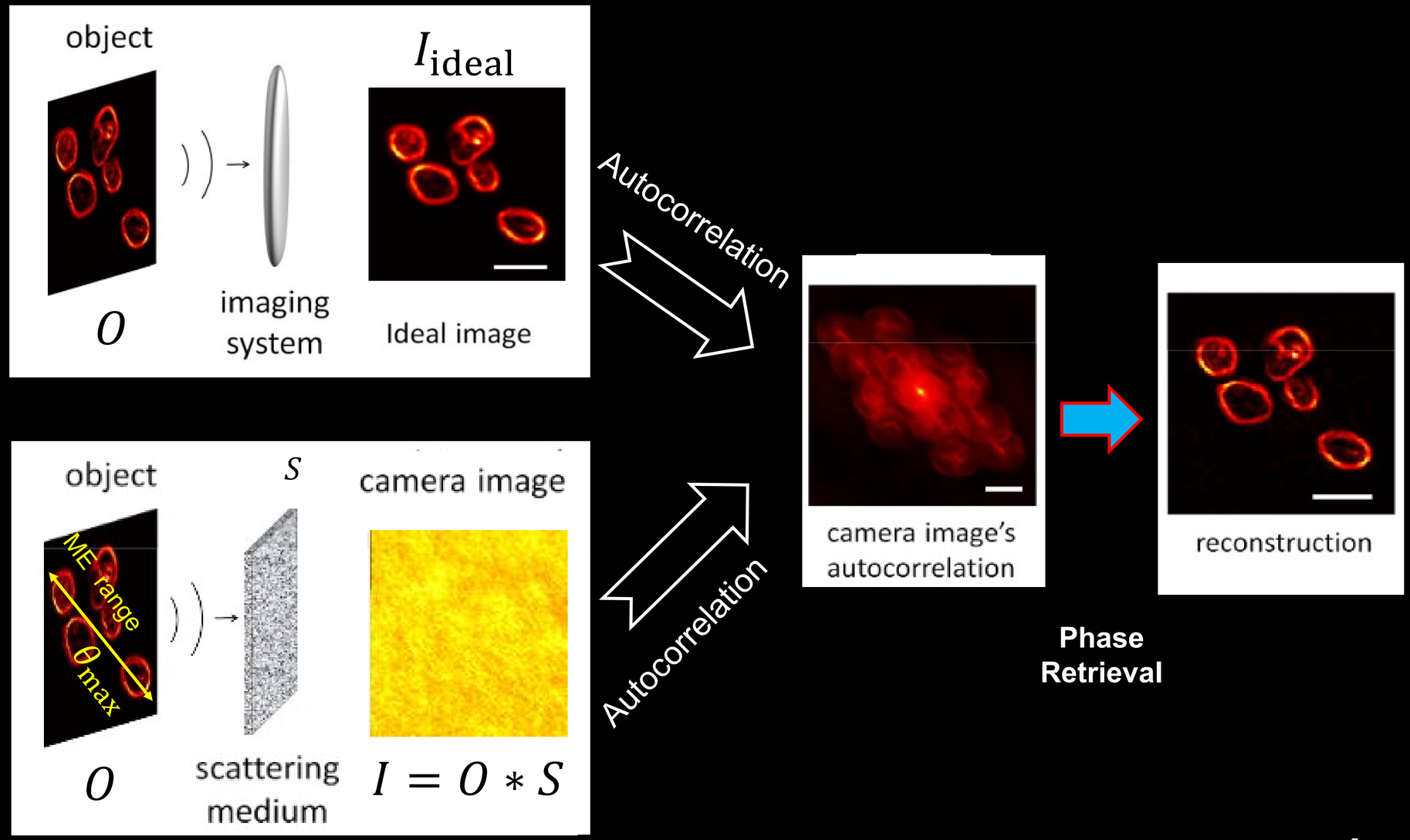
[Katz et al., 2014]

Use the memory effect to image through scattering



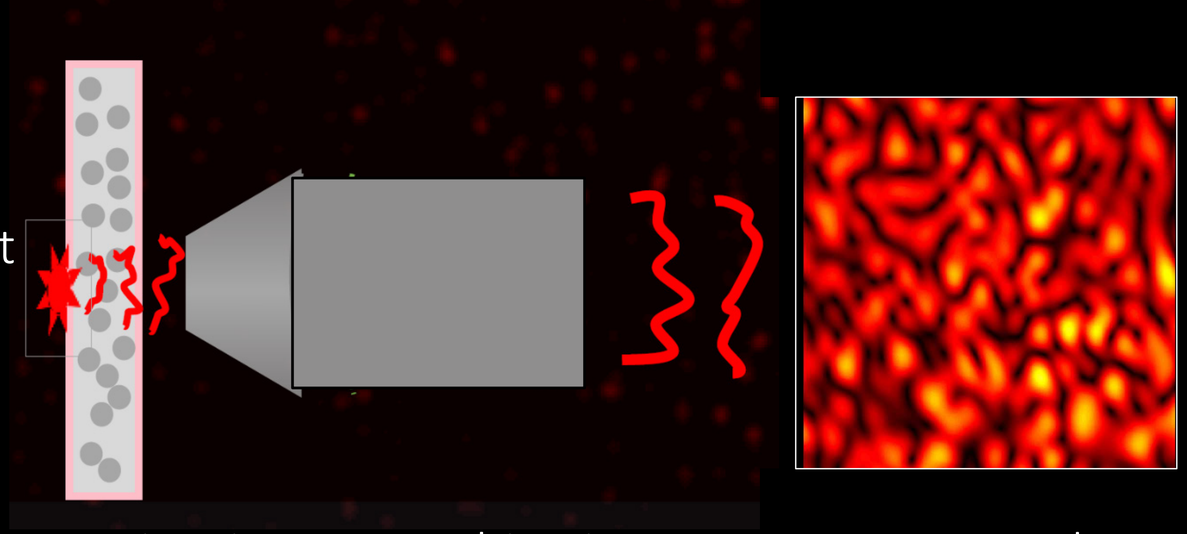
[Katz et al., 2014]

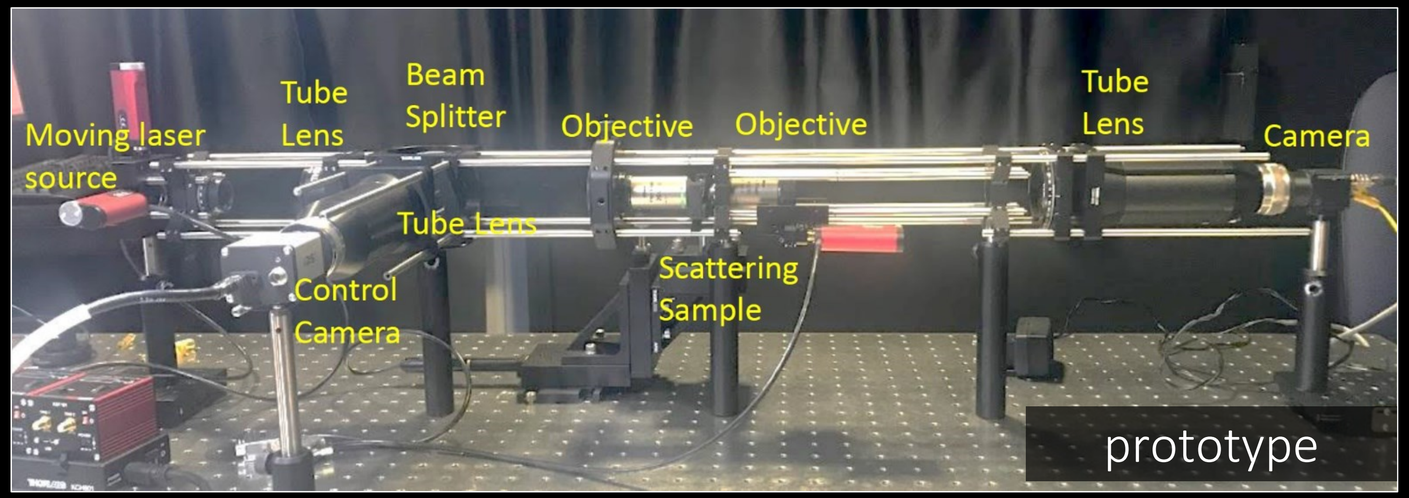
Use the memory effect to image through scattering



[Katz et al., 2014]

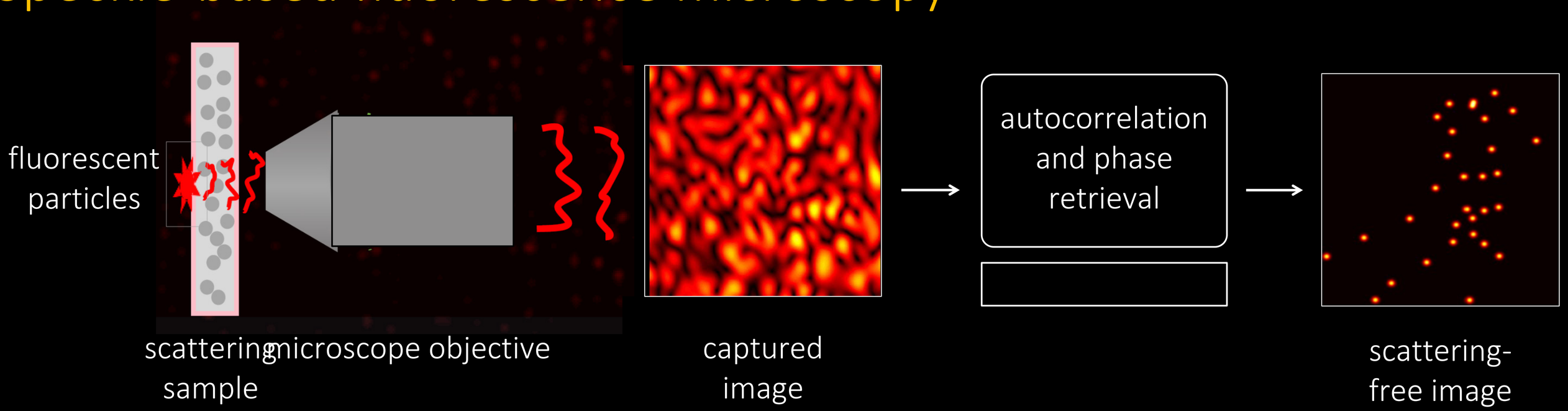
Speckle-based fluorescence microscopy

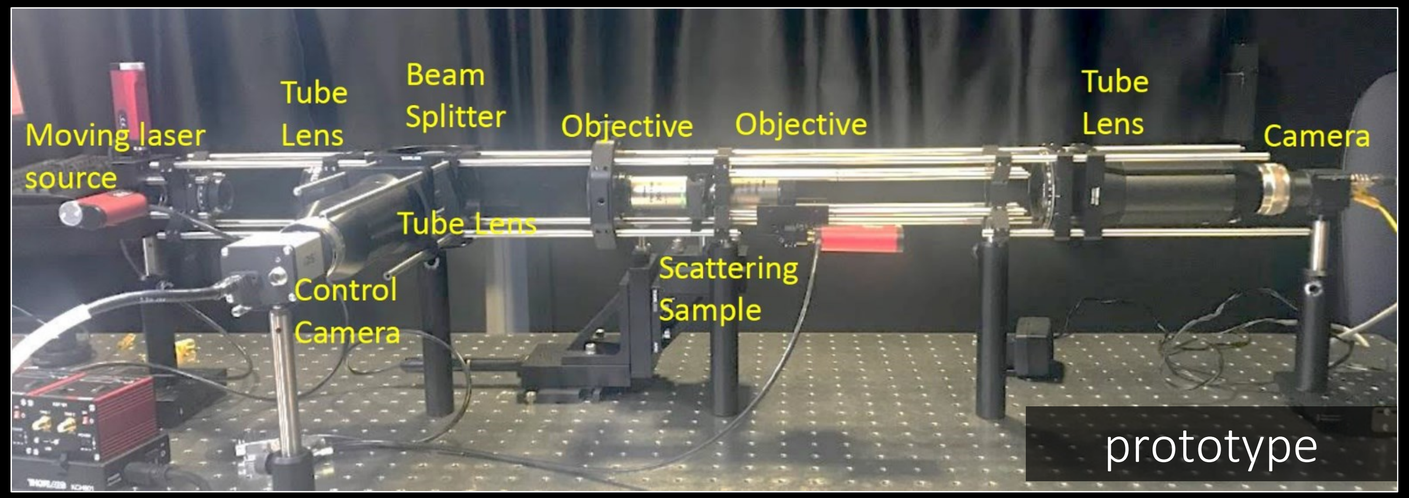




[PIs: Gkioulekas, Levin]

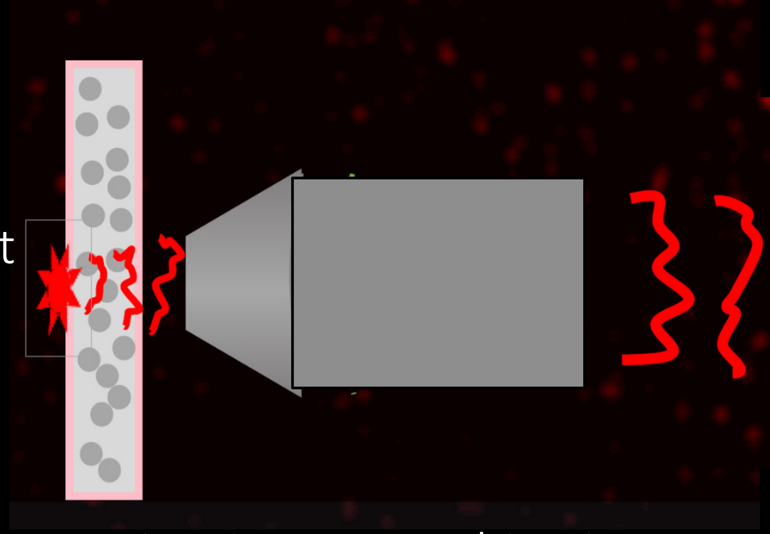
Speckle-based fluorescence microscopy

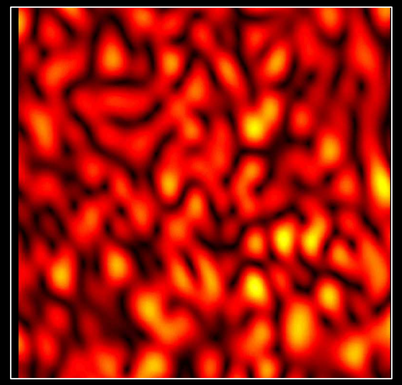




[PIs: Gkioulekas, Levin]

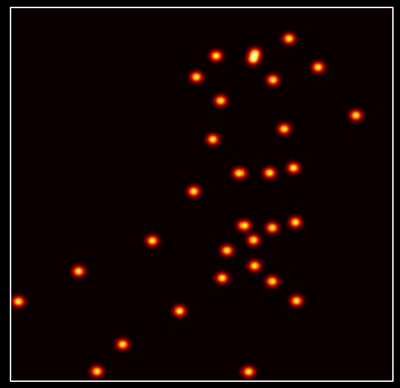
Speckle-based fluorescence microscopy



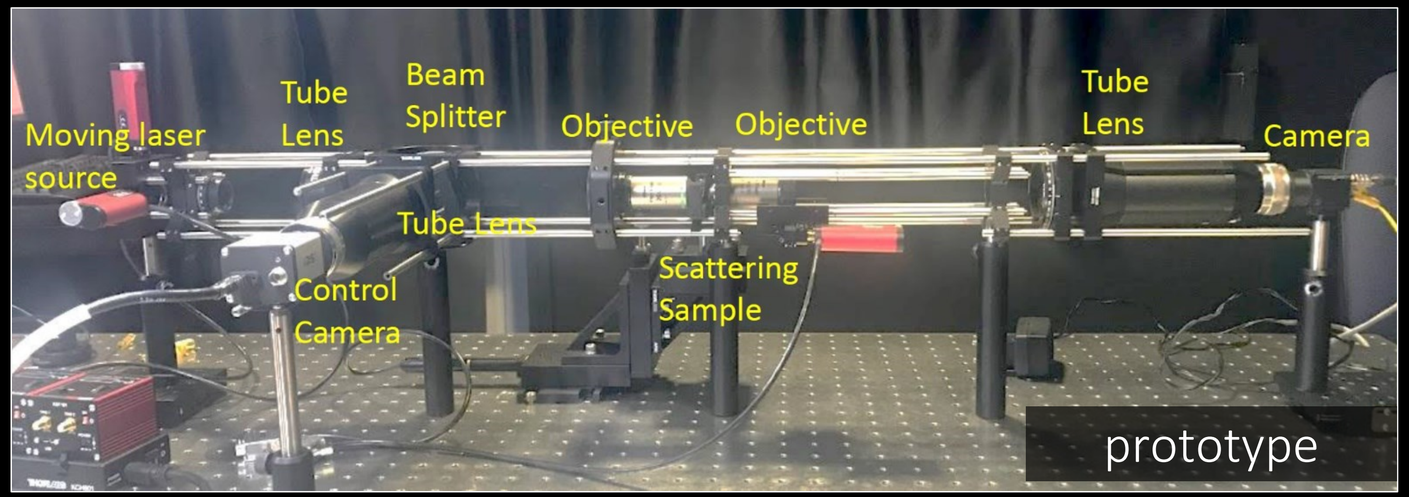


captured image

autocorrelation and phase retrieval



scattering-free image



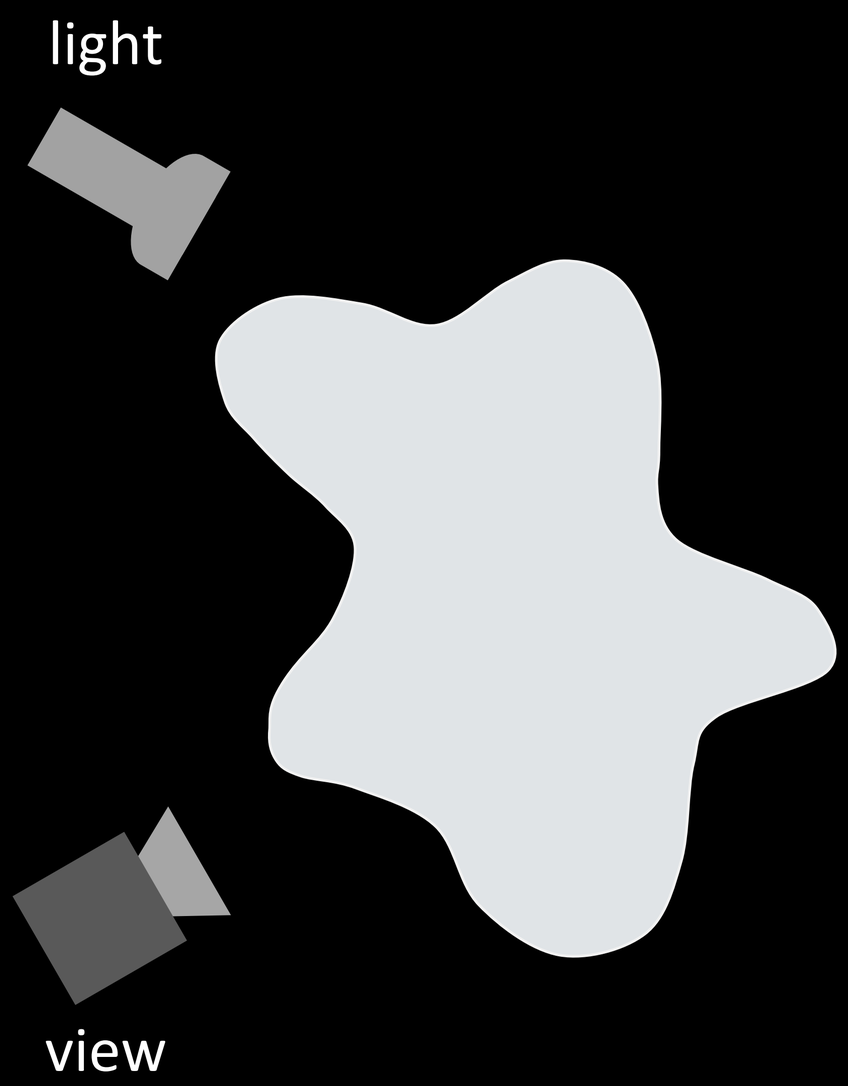
[PIs: Gkioulekas, Levin]

Performance strongly depends on:

- speckle statistics
- image priors
- tissue parameters

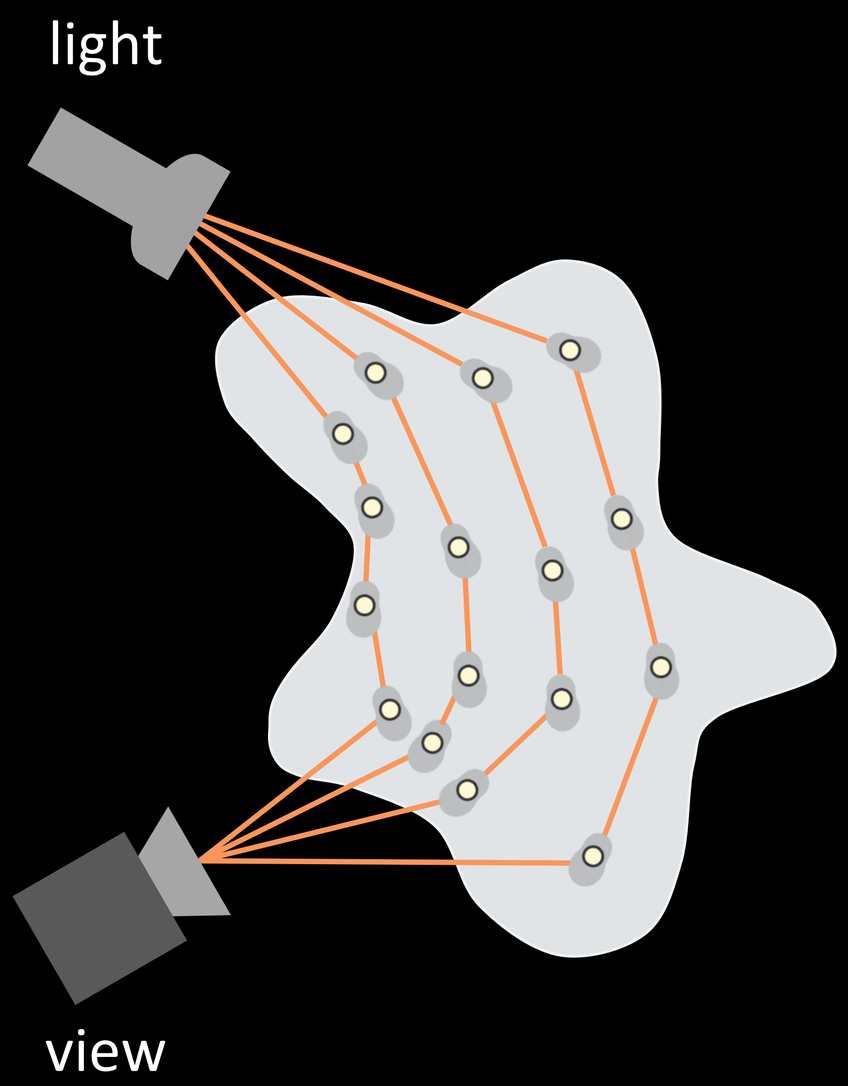
Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$



[Bar et al., 2019]

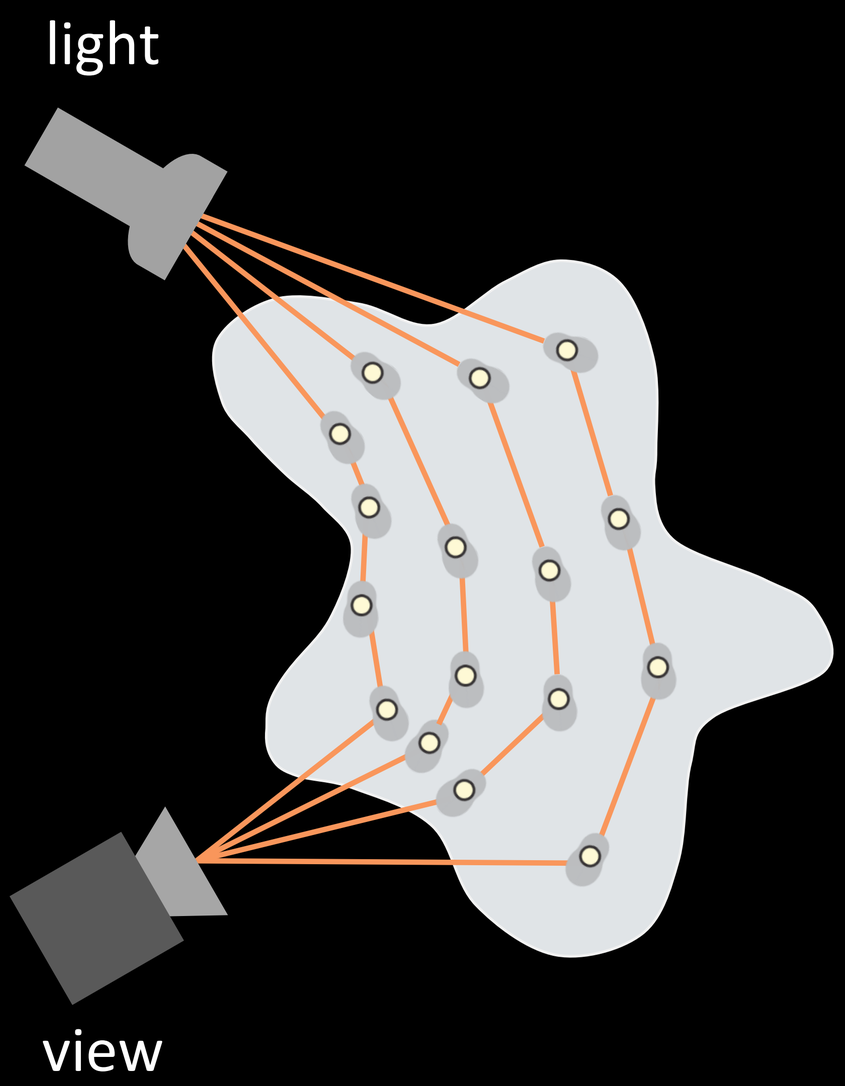
Recap: Monte Carlo (volumetric) rendering



$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

[Bar et al., 2019]

Recap: Monte Carlo (volumetric) rendering



$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

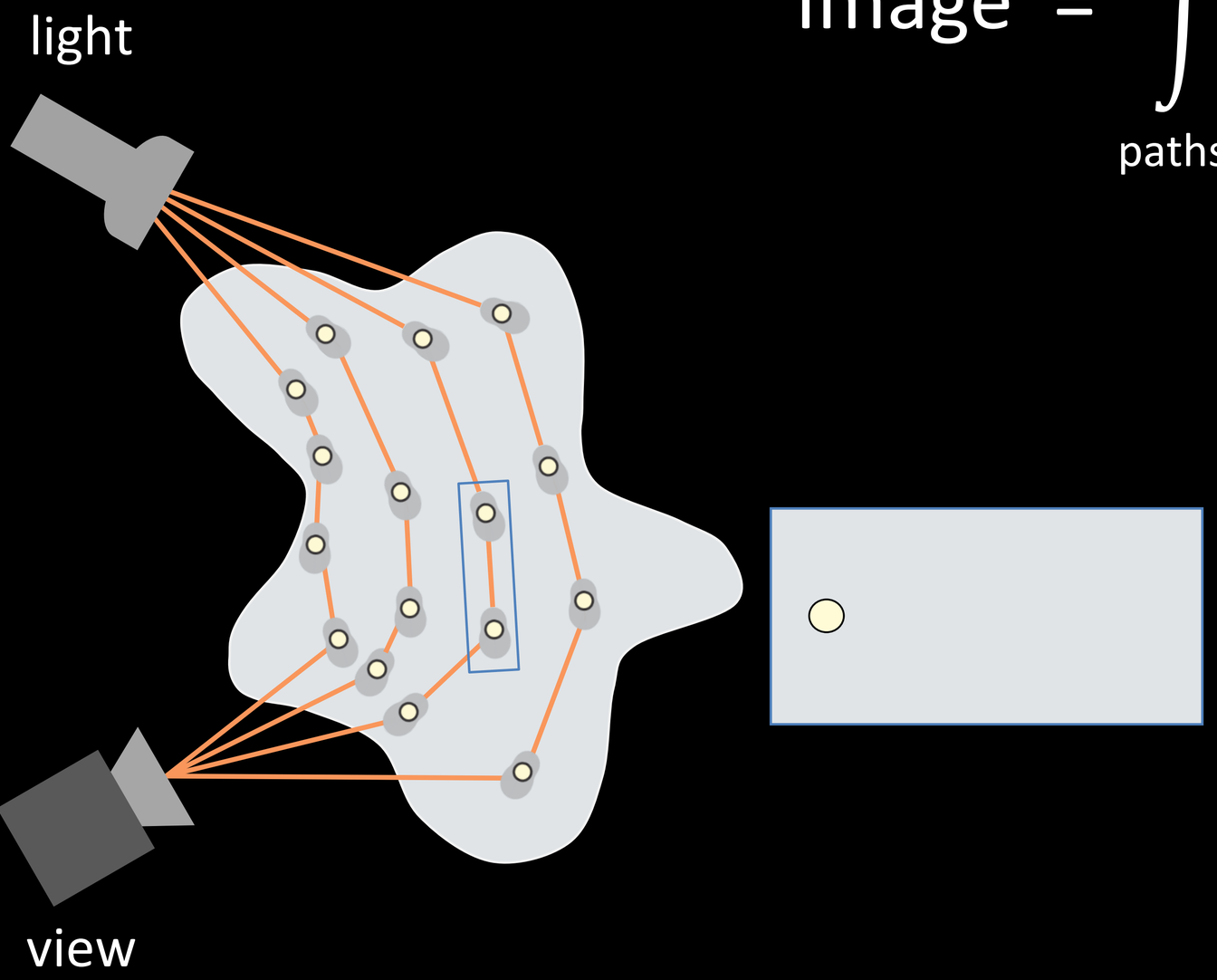
Path contribution, depends on the scattering material

[Bar et al., 2019]

Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

Path contribution, depends on the scattering material

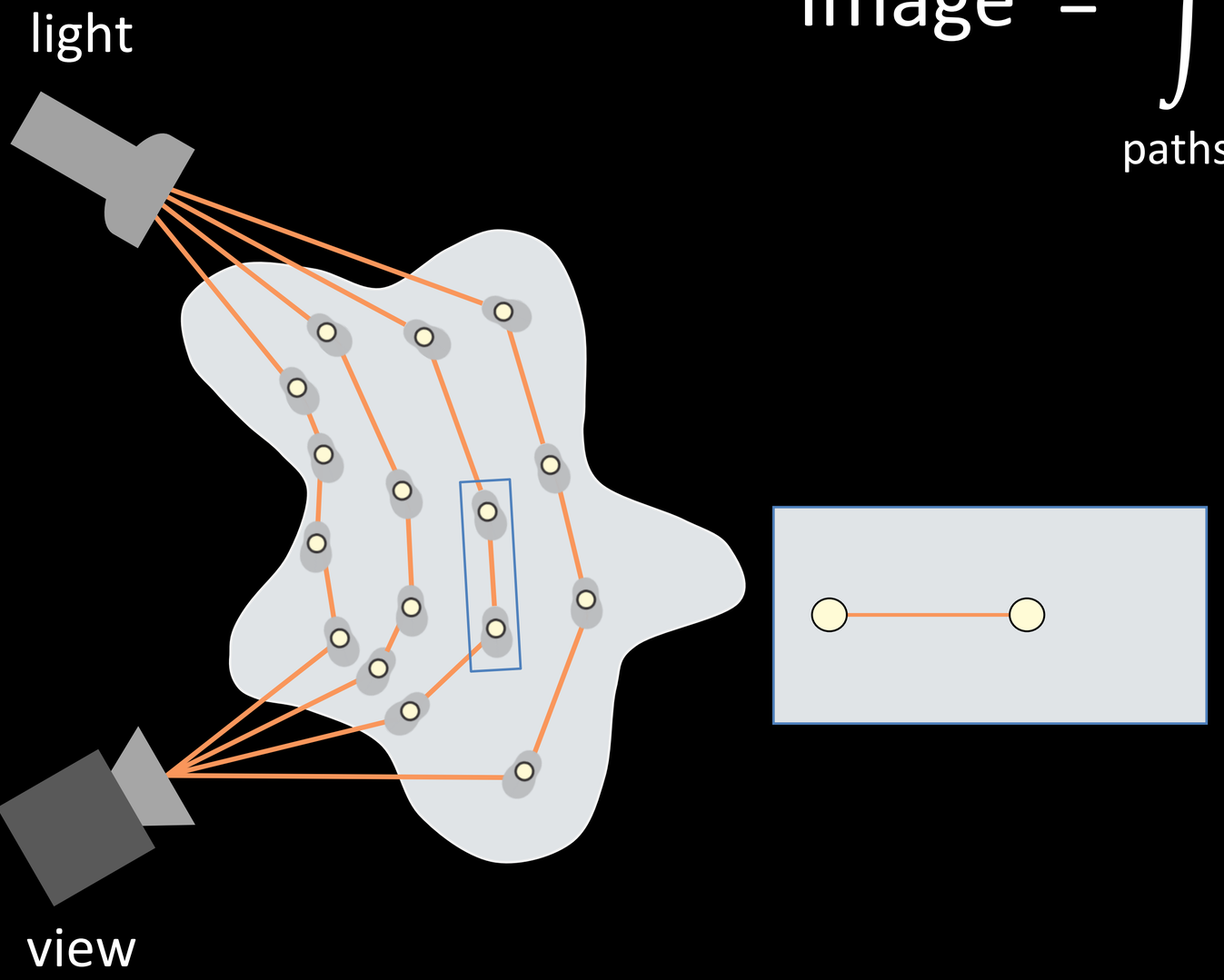


[Bar et al., 2019]

Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

Path contribution, depends on the scattering material



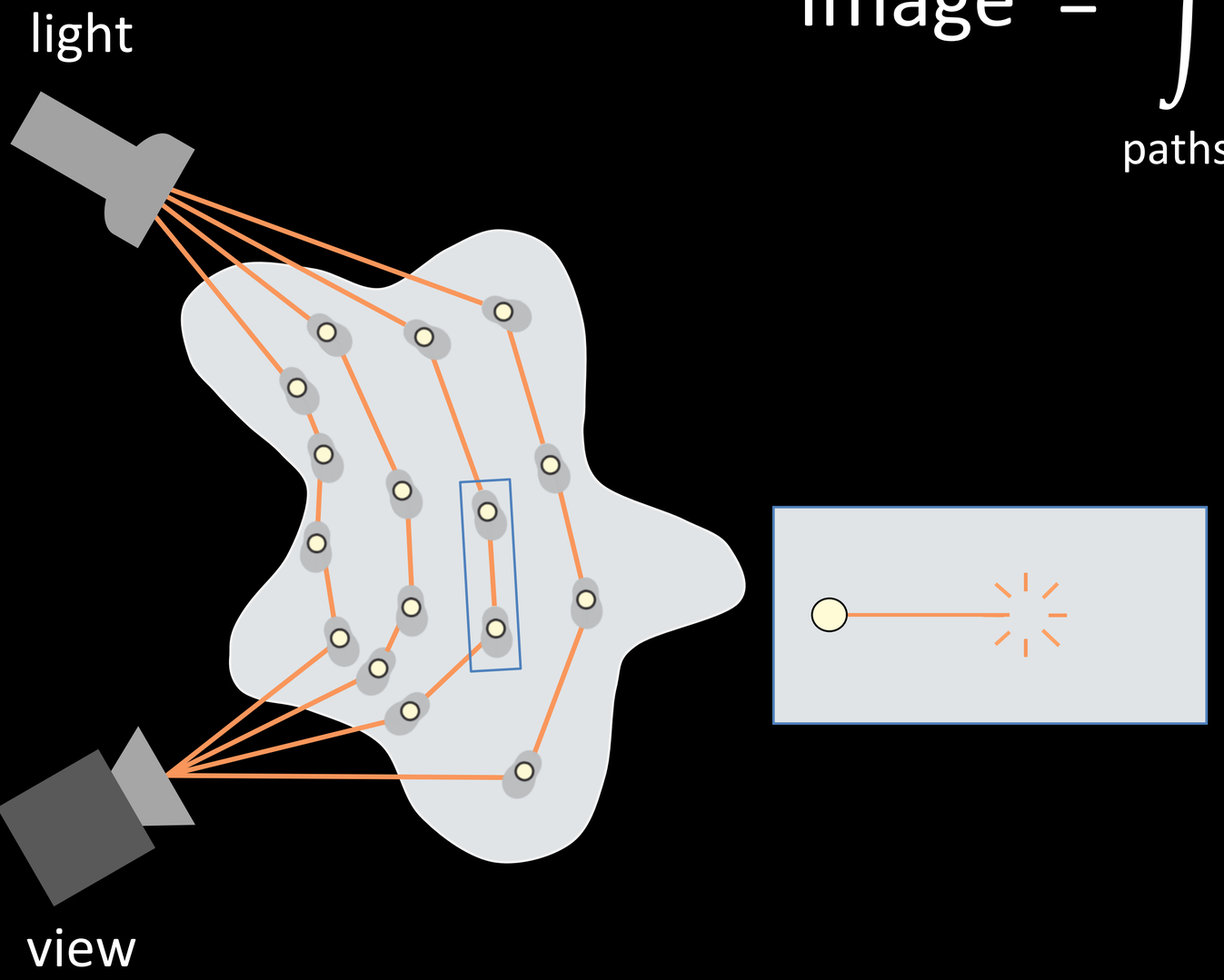
volumetric density
(extinction coefficient) σ

[Bar et al., 2019]

Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

Path contribution, depends on the scattering material



volumetric density (extinction coefficient) σ

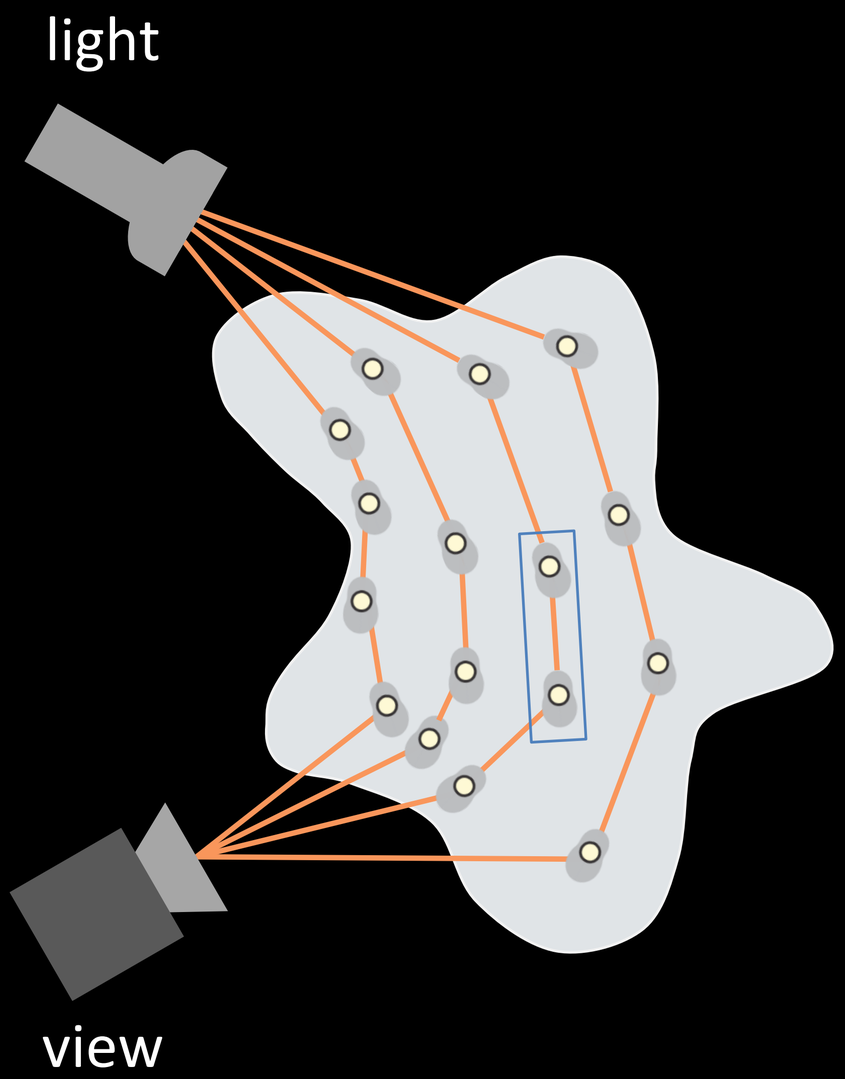
scattering albedo a

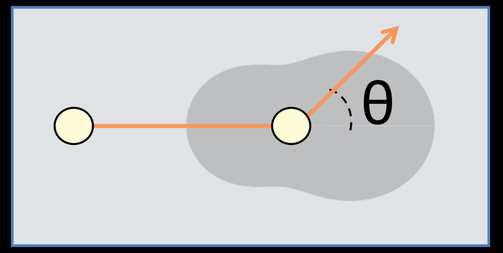
[Bar et al., 2019]

Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

Path contribution, depends on the scattering material





volumetric density (extinction coefficient) σ

scattering albedo a

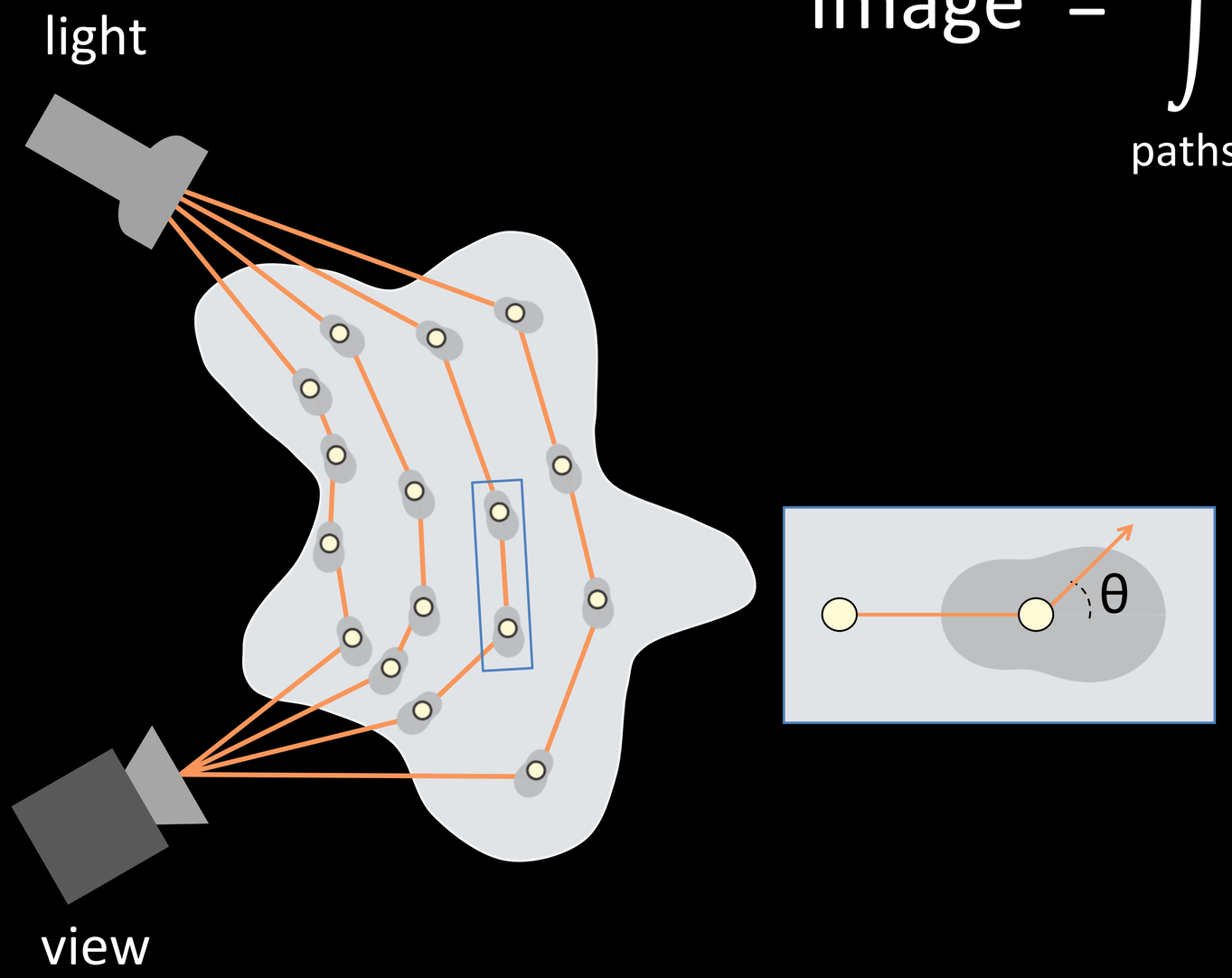
phase function p_θ

[Bar et al., 2019]

Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

Path contribution, depends on the scattering material



$$\text{material} = \begin{bmatrix} \sigma \\ a \\ p_\theta \end{bmatrix}$$

[Bar et al., 2019]

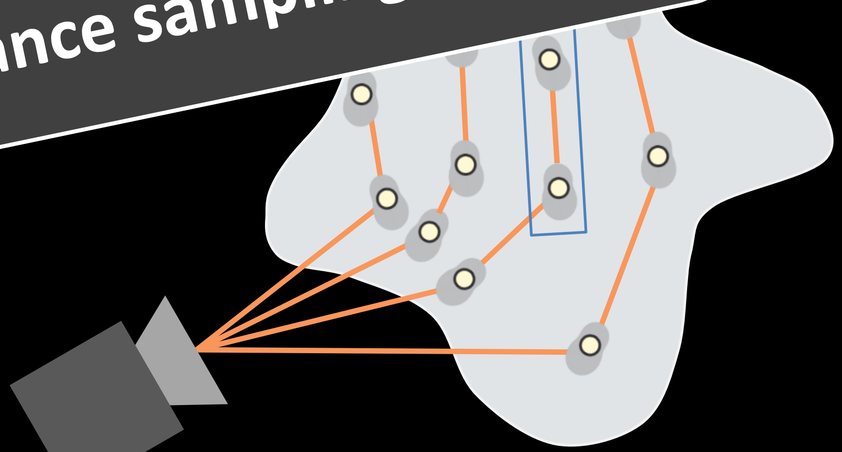
Recap: Monte Carlo (volumetric) rendering

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

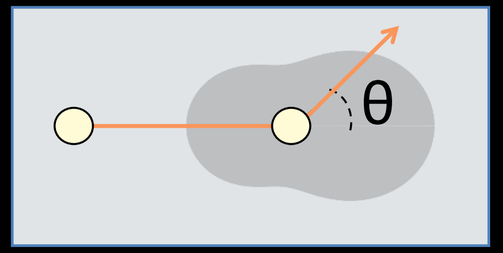
Path contribution, depends on the scattering material

light

To compute efficiently: importance sampling of paths



view

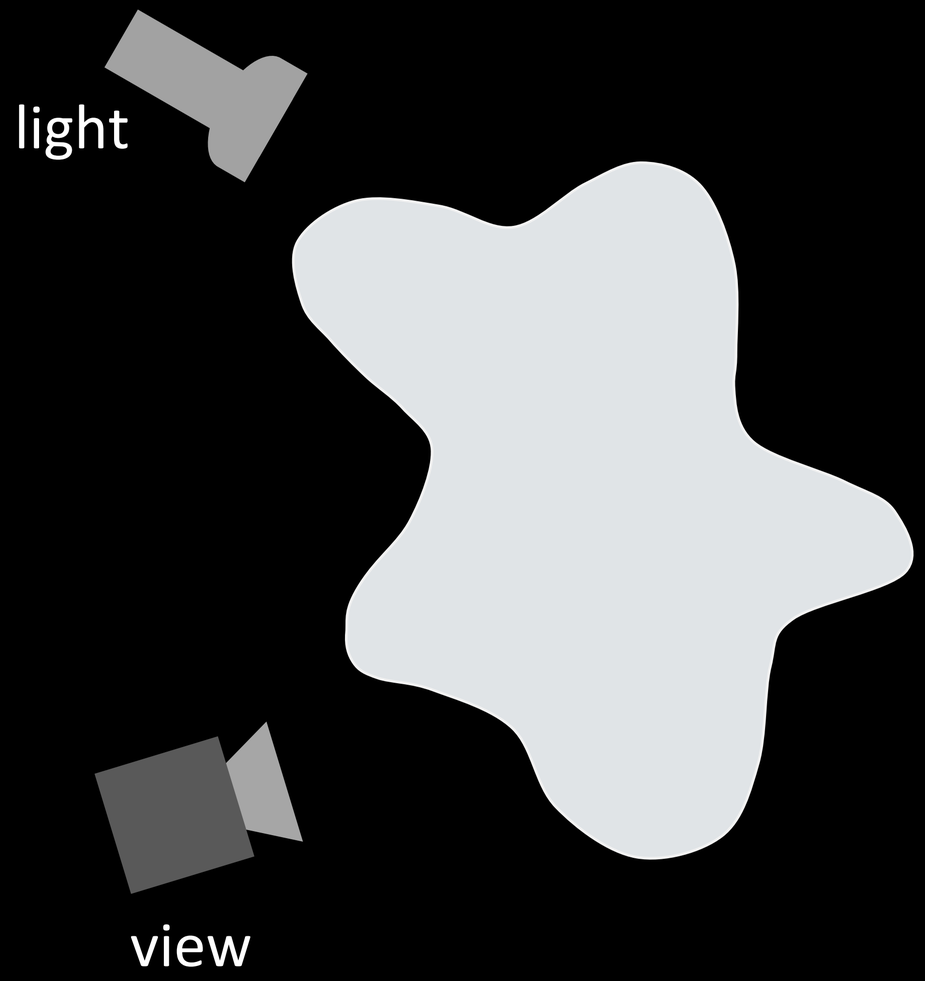


$$\text{material} = \begin{bmatrix} \sigma \\ a \\ p_\theta \end{bmatrix}$$

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images

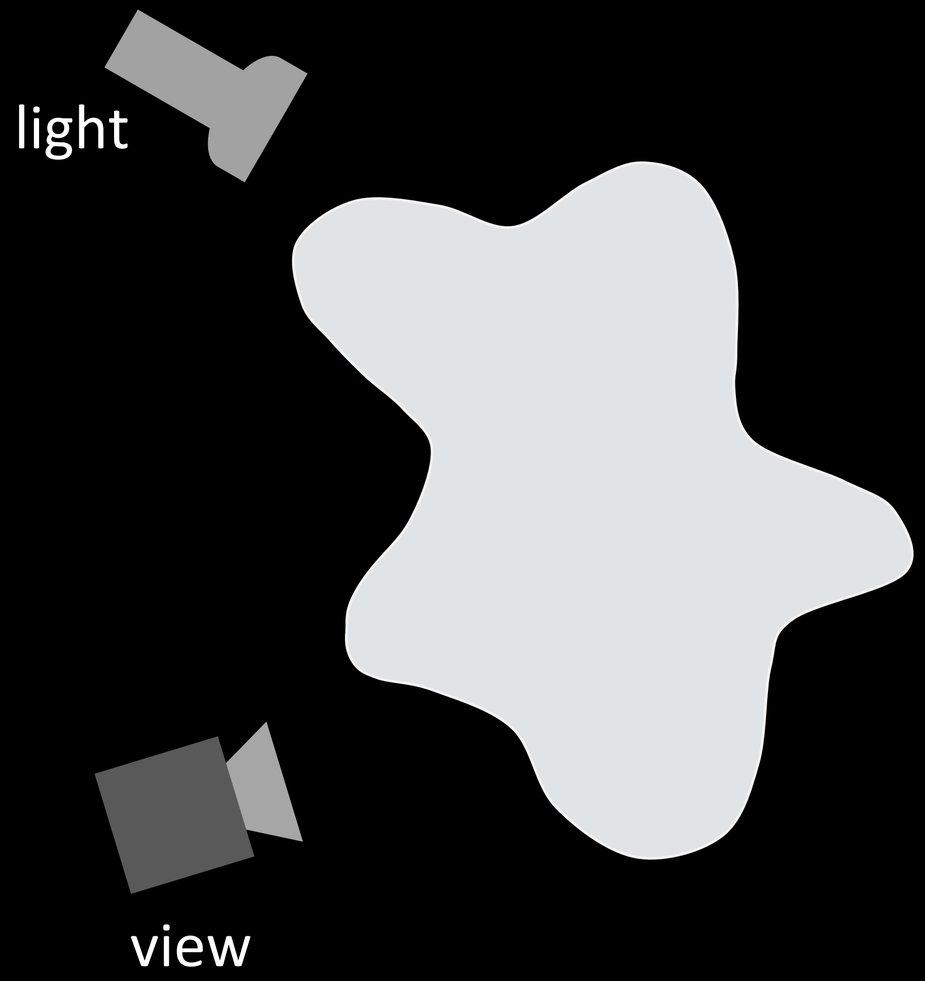
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$



[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images

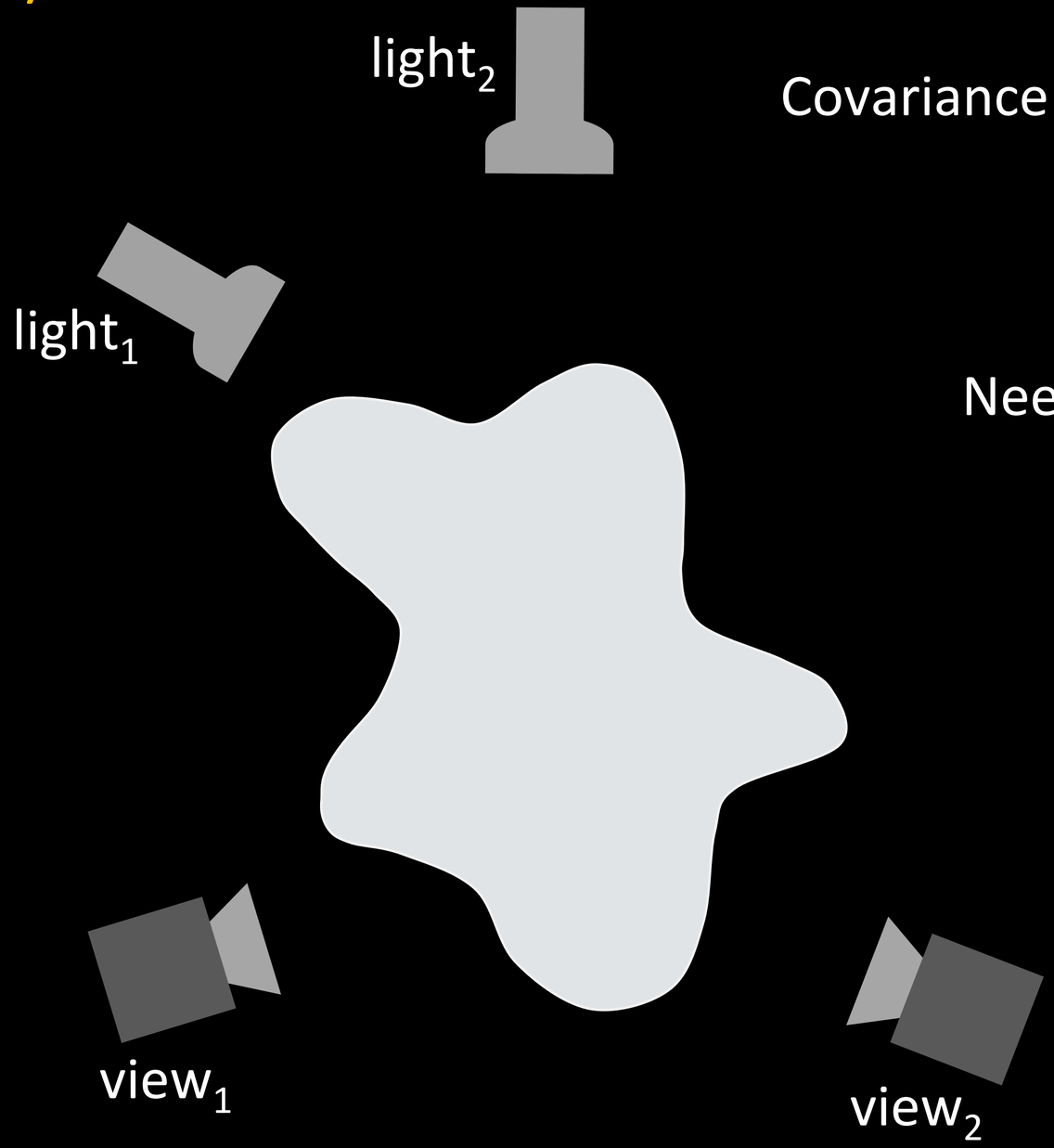
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$



Need to consider products of ***pairs*** of paths

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images

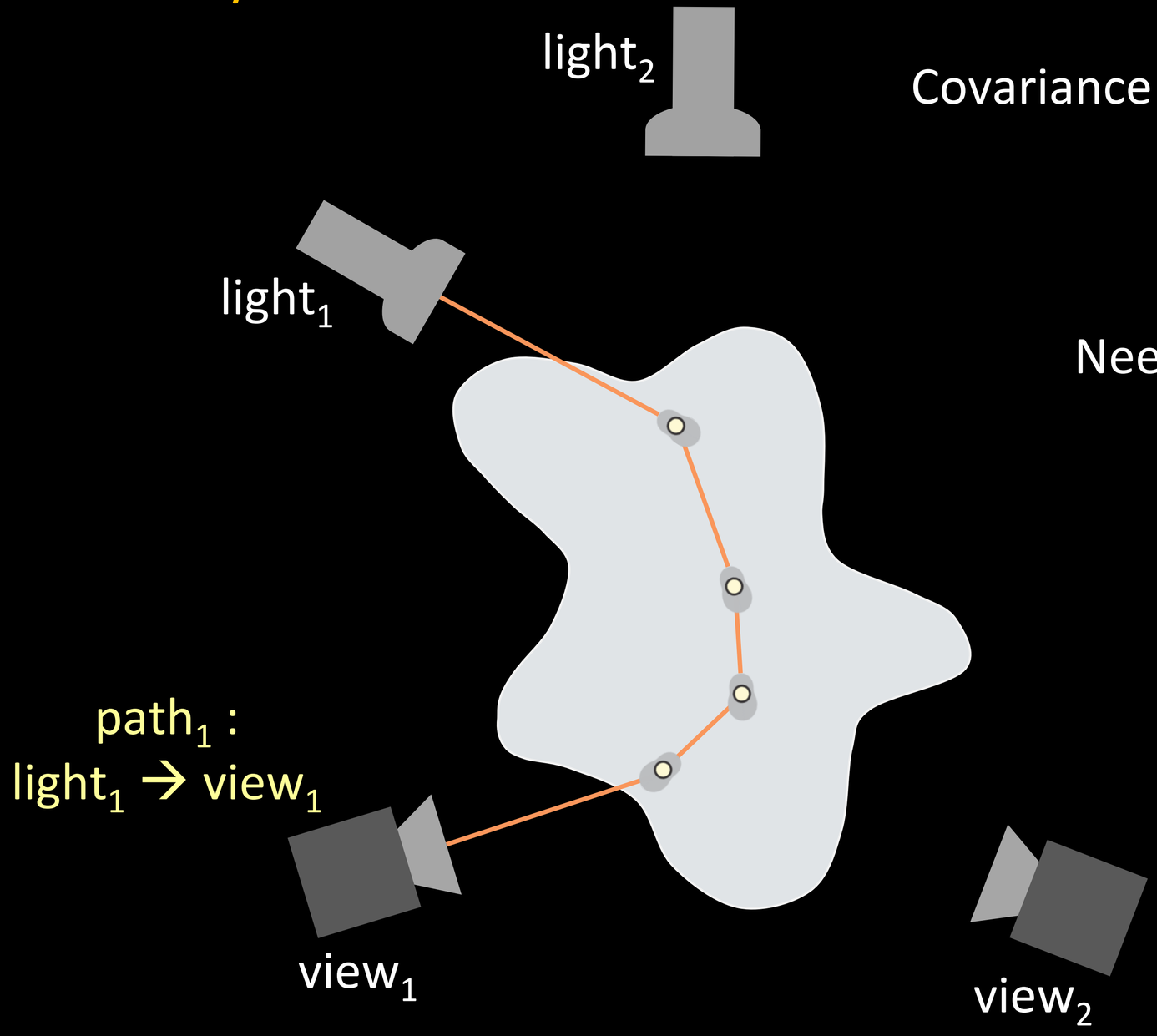


$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

Need to consider products of ***pairs*** of paths

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images

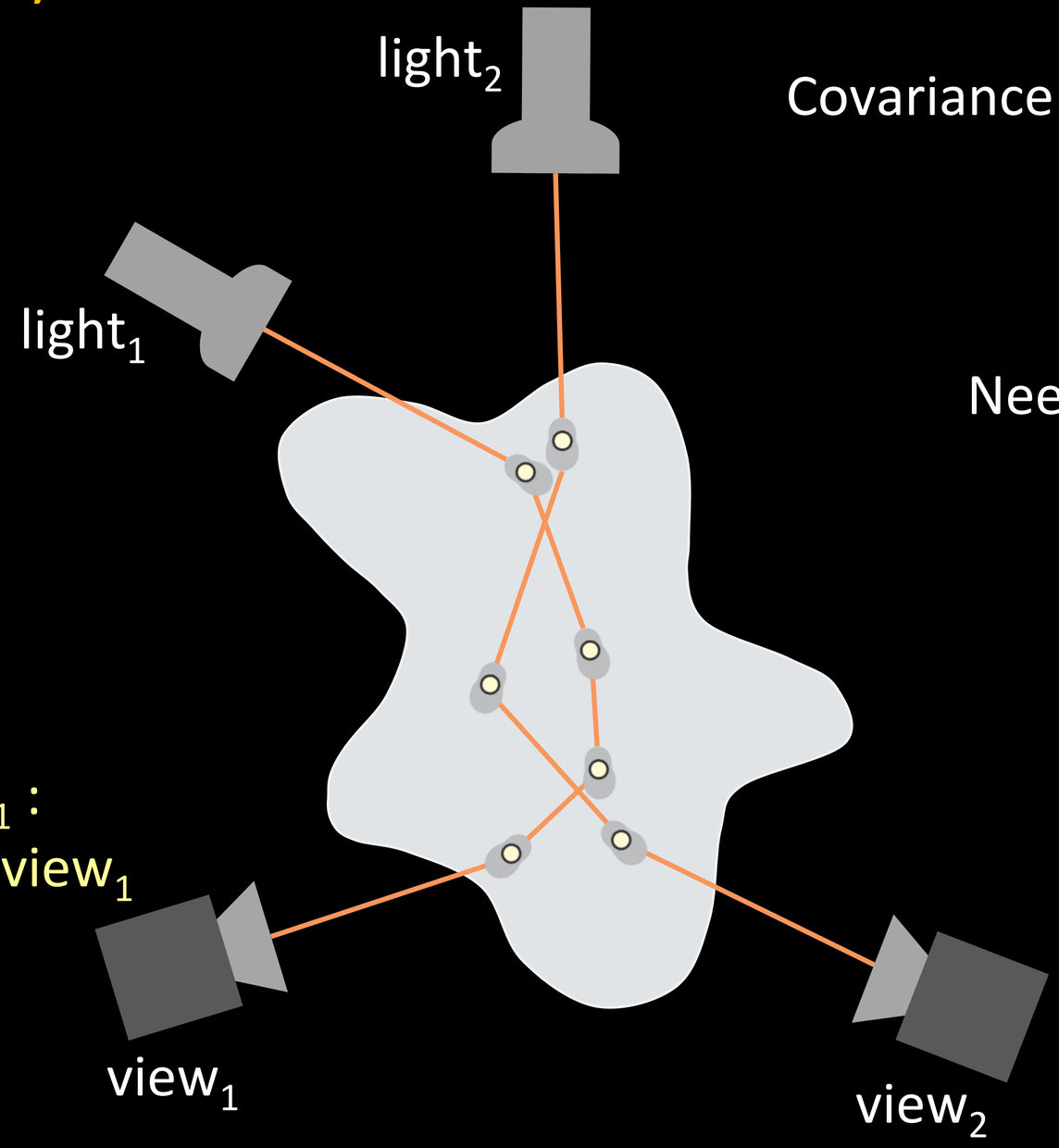


$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

Need to consider products of ***pairs*** of paths

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images

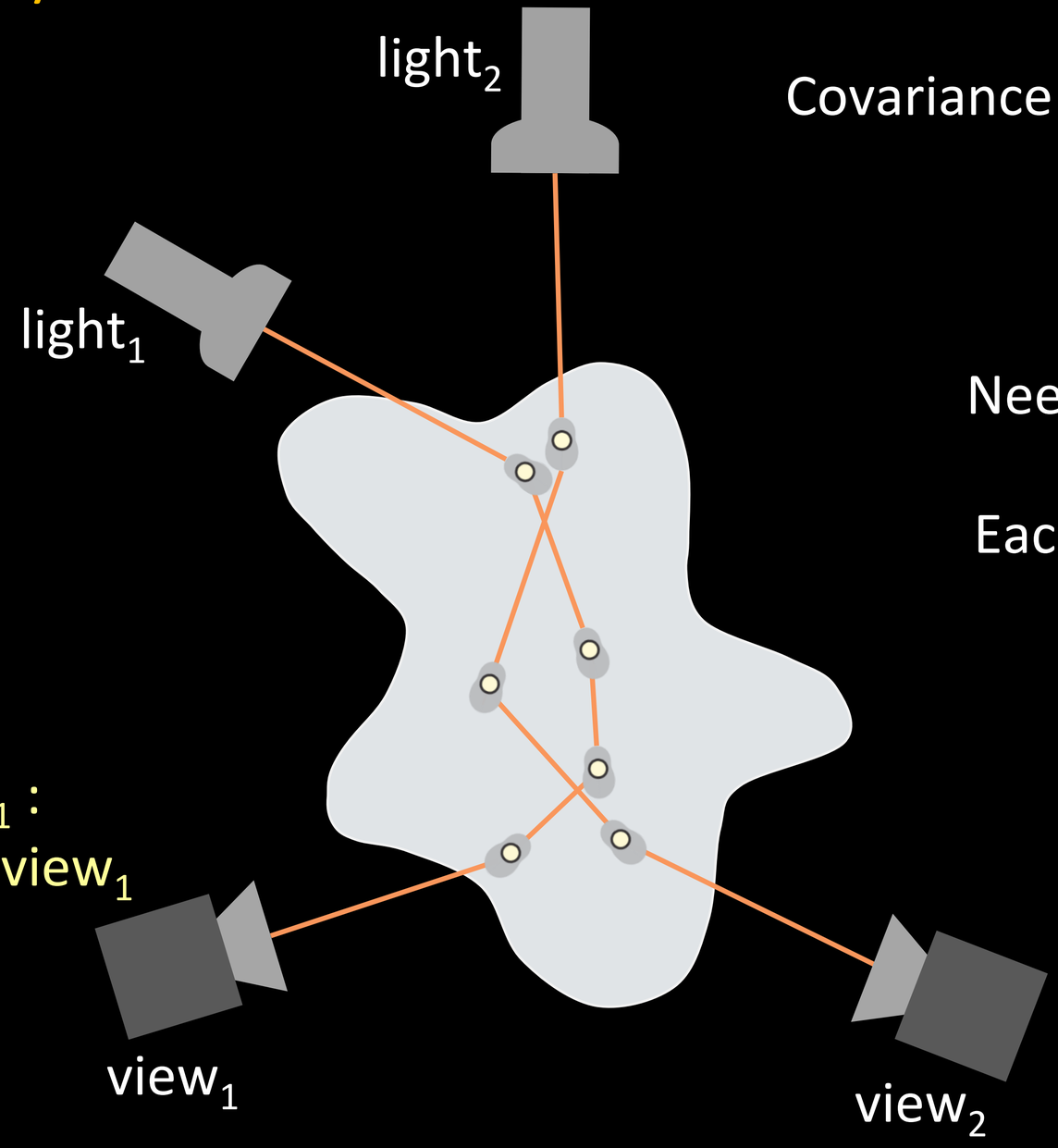


$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

Need to consider products of ***pairs*** of paths

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

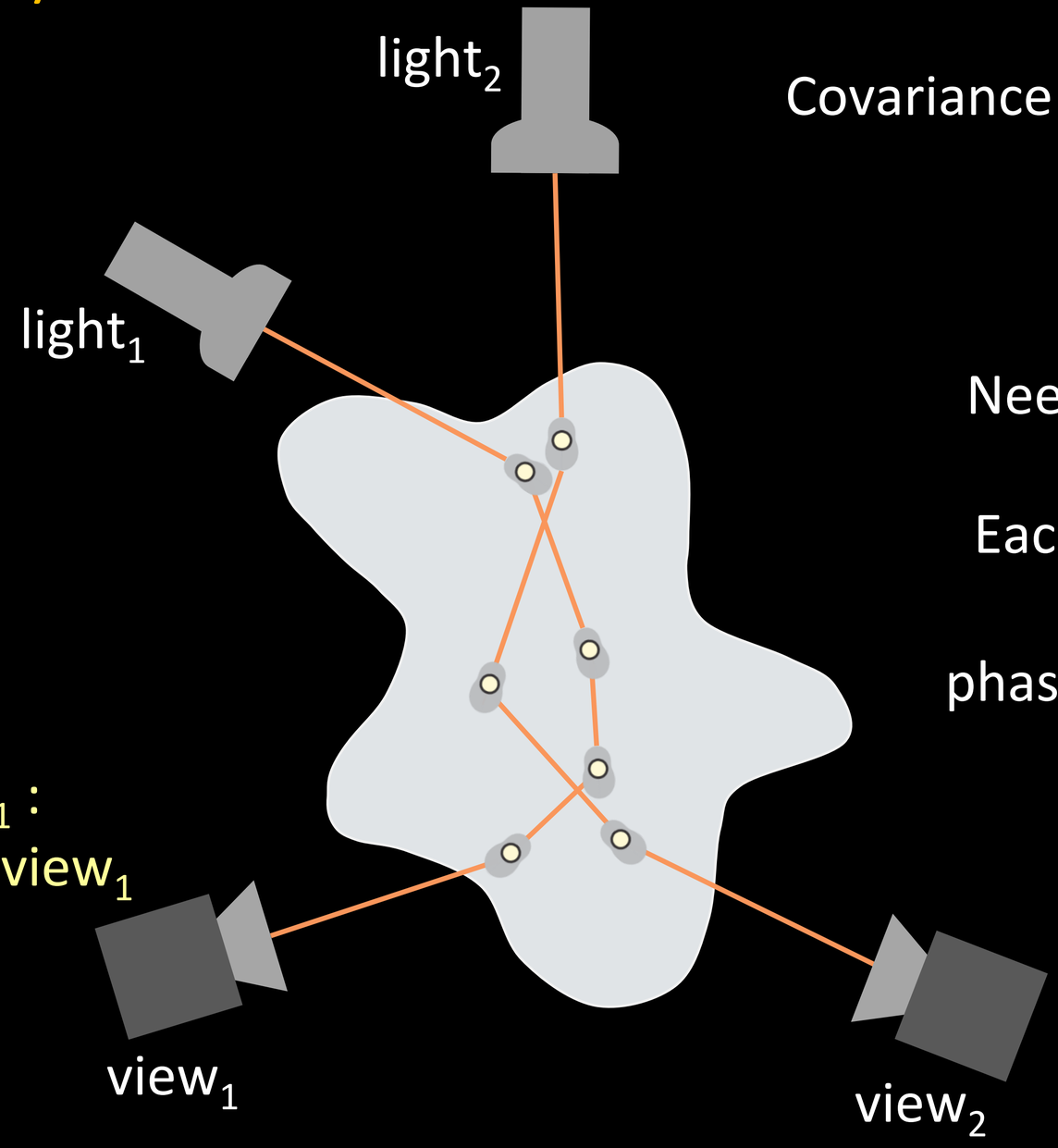
$$u = |u|\, e^{i \cdot \text{phase}}$$

Need to consider products of ***pairs*** of paths

Each path contributes a complex number u

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

$$u = |u|\, e^{i \cdot \text{phase}}$$

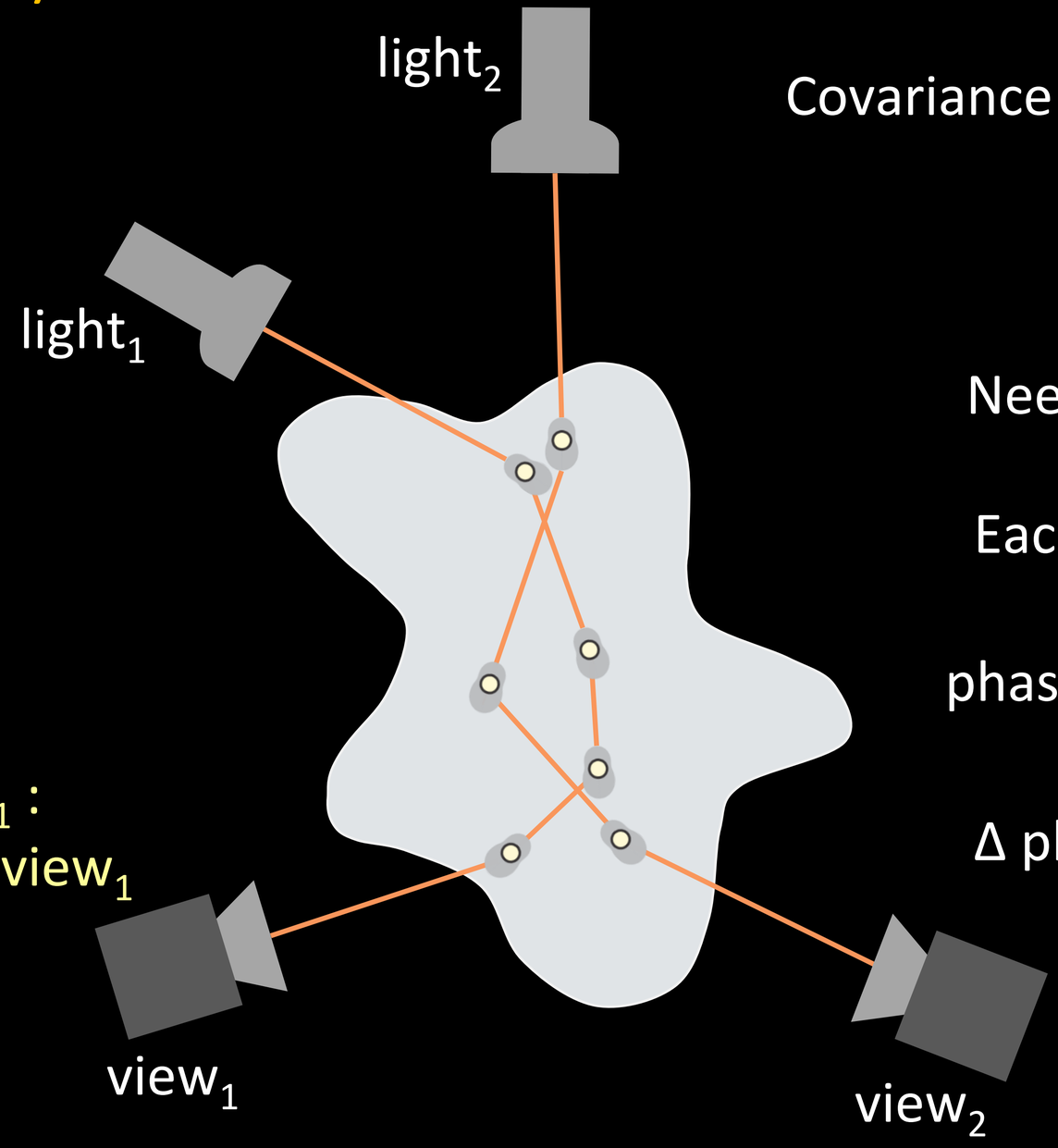
Need to consider products of ***pairs*** of paths

Each path contributes a complex number u

phase ∝ Length (path)

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

$$u = |u|\, e^{i \cdot \text{phase}}$$

Need to consider products of ***pairs*** of paths

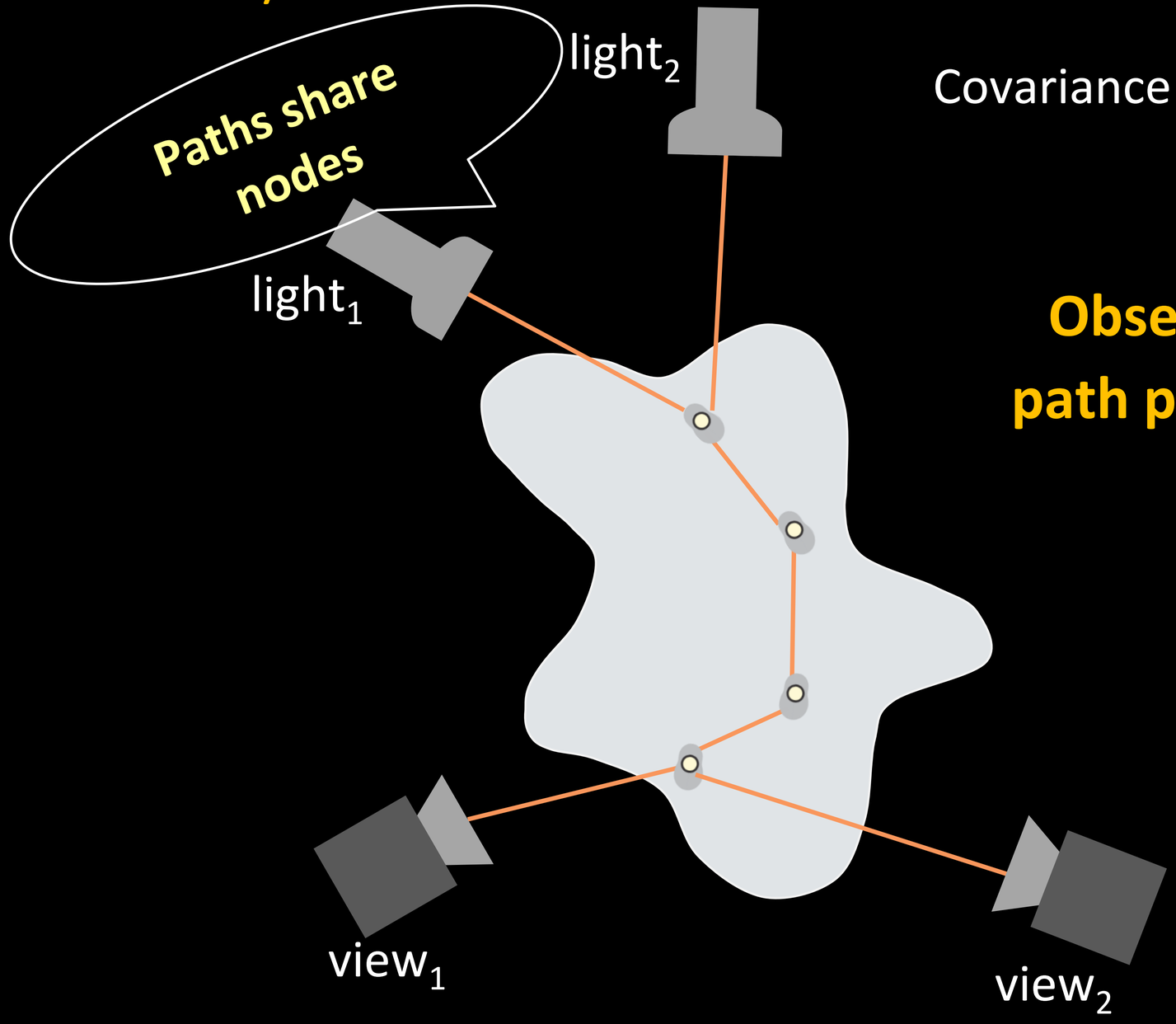
Each path contributes a complex number u

phase ∝ Length (path)

Δ phase ∝ Length (path$_1$) - Length (path$_2$)

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

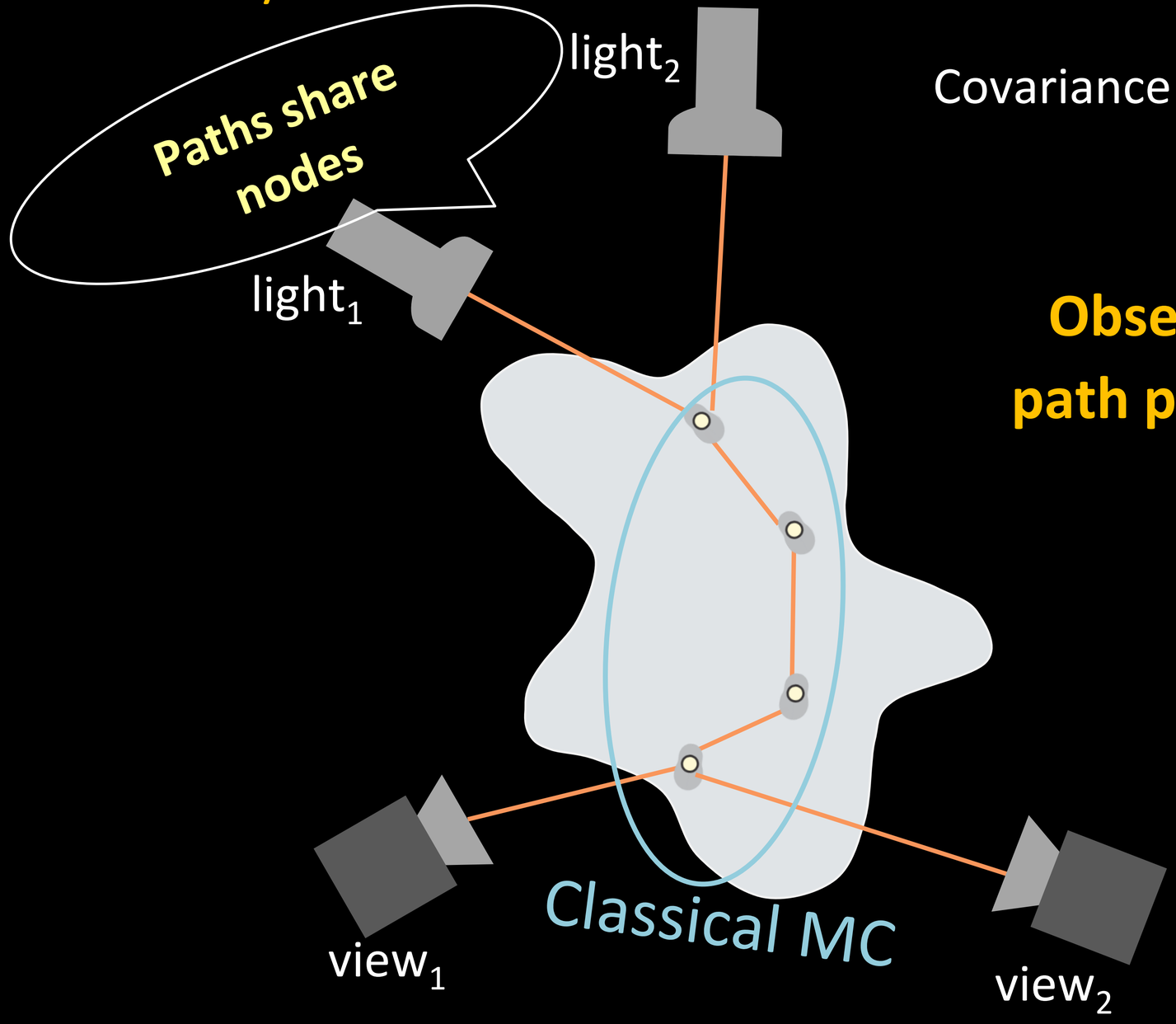
Observation: need to consider only path pairs that share the *same* nodes (except start and end)

⇩

All other path pairs are averaged out in the integration

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

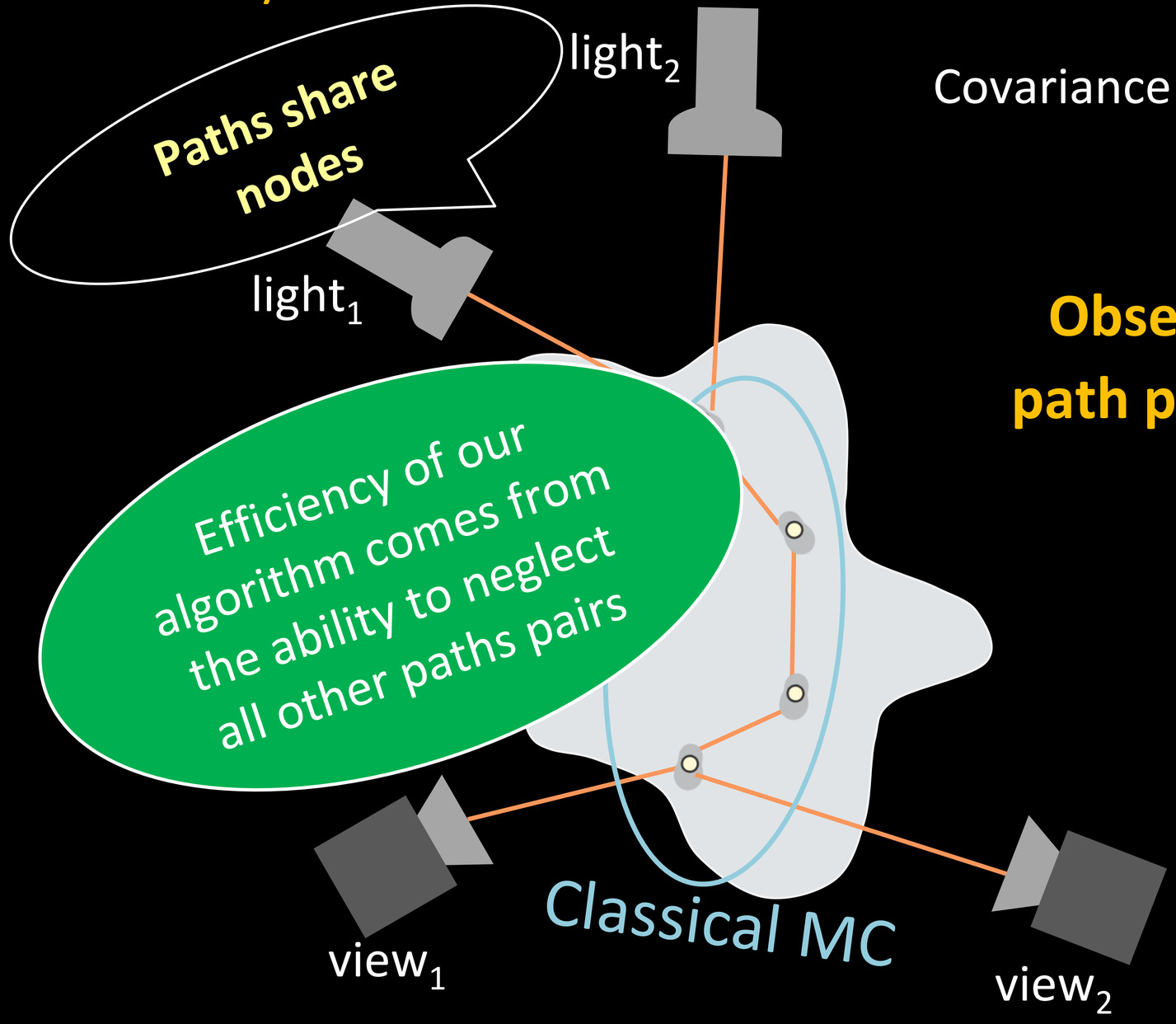
Observation: need to consider only path pairs that share the *same* nodes
(except start and end)

⇩

All other path pairs are averaged out in the integration

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

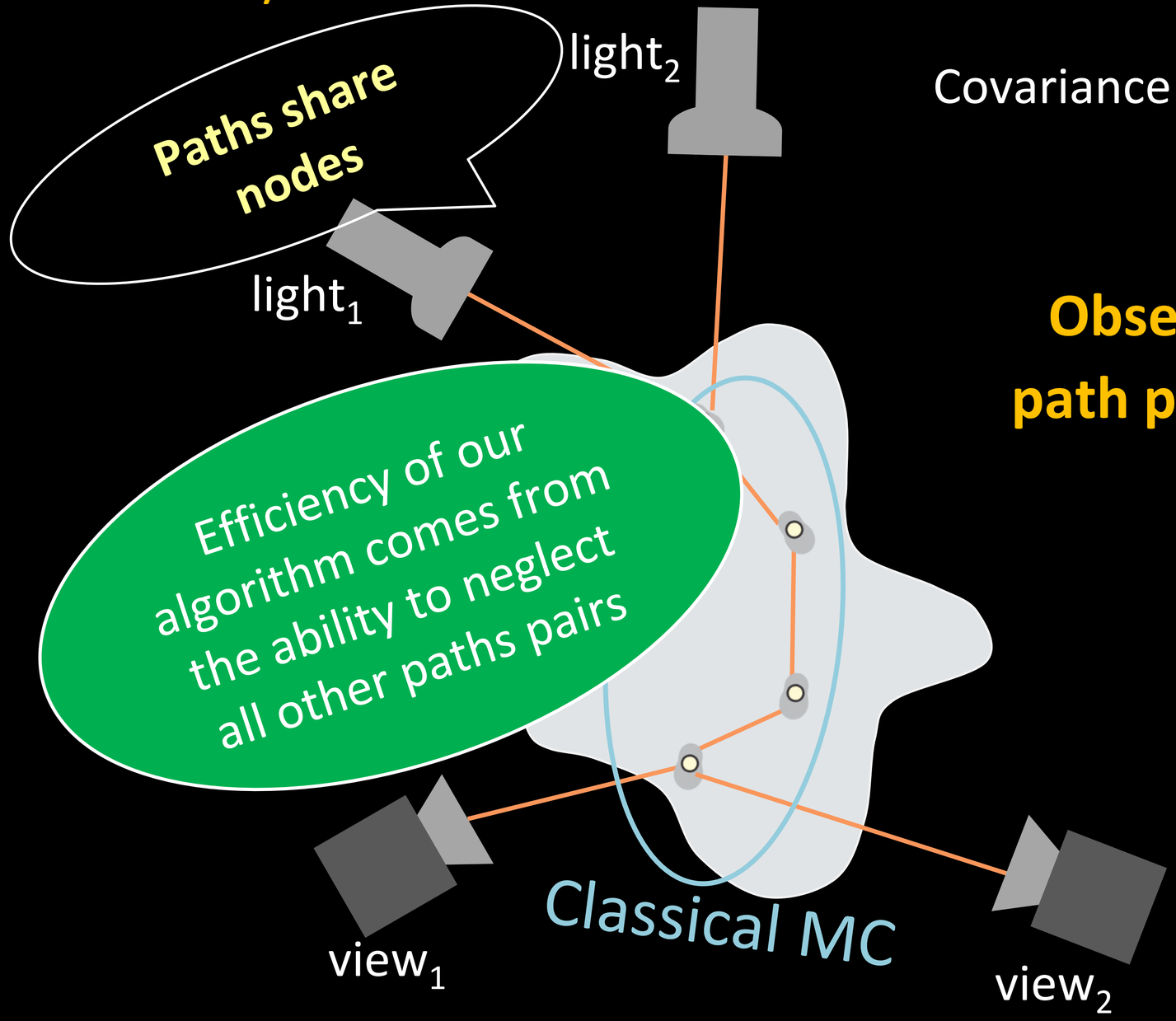
Observation: need to consider only path pairs that share the *same* nodes
(except start and end)

⇩

All other path pairs are averaged out in the integration

[Bar et al., 2019]

Memory effect: simulate *covariance* of speckle images

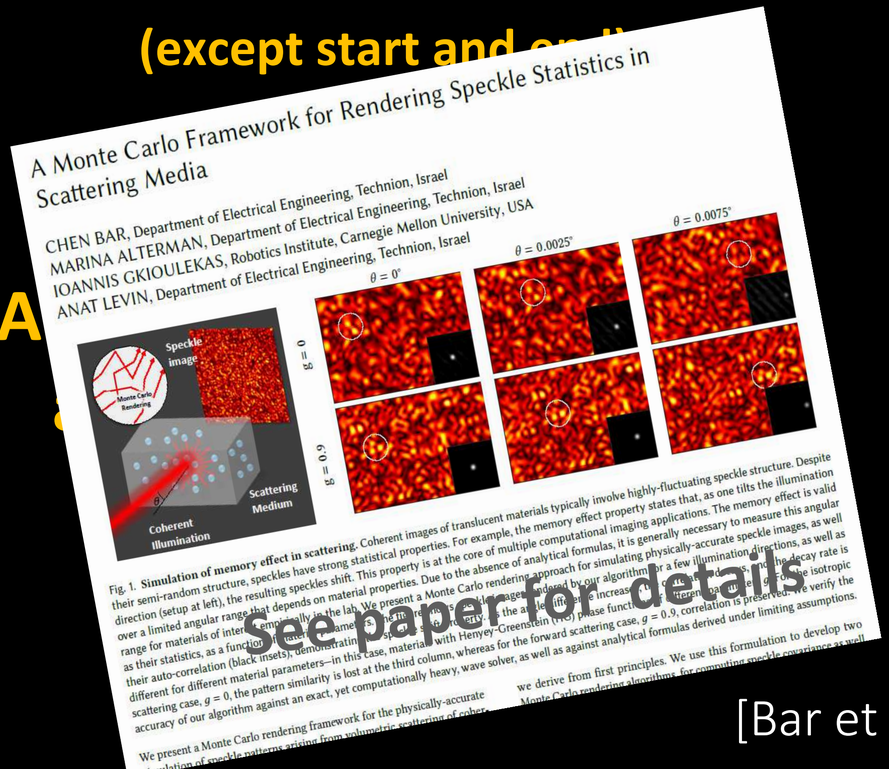


$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

Observation: need to consider only path pairs that share the *same* nodes

(except start and end)

A


A Monte Carlo Framework for Rendering Speckle Statistics in Scattering Media

CHEN BAR, Department of Electrical Engineering, Technion, Israel
MARINA ALTERMAN, Department of Electrical Engineering, Technion, Israel
IOANNIS GKIOULEKAS, Robotics Institute, Carnegie Mellon University, USA
ANAT LEVIN, Department of Electrical Engineering, Technion, Israel

$\theta = 0°$ $\theta = 0.0025°$ $\theta = 0.0075°$

Speckle image
Monte Carlo Rendering
Coherent Illumination
Scattering Medium
$g = 0$
$g = 0.9$

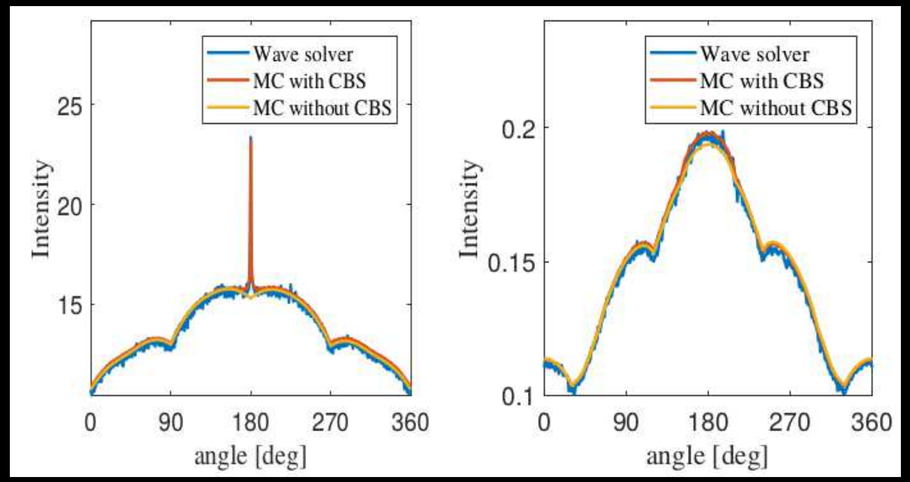
Fig. 1. **Simulation of memory effect in scattering.** Coherent images of translucent materials typically involve highly-fluctuating speckle structure. Despite their semi-random structure, speckles have strong statistical properties. For example, the memory effect property states that, as one tilts the illumination direction (setup at left), the resulting speckles shift. This property is at the core of multiple computational imaging applications. The memory effect is valid over a limited angular range that depends on material properties. Due to the absence of analytical formulas, it is generally necessary to measure this angular range for materials of interest empirically in the lab. We present a Monte Carlo rendering approach for simulating physically-accurate speckle images, as well as their statistics, as a function of material parameters—in this case, material with Henyey-Greenstein (HG) phase function parameter g. The decay rate is their auto-correlation (black insets); demonstrating the increasingly narrower angular range of the memory effect for different material parameters—in this case, material with Henyey-Greenstein (HG) phase function parameter g. The decay rate is different for different material parameters—in this case, material with Henyey-Greenstein (HG) phase function parameter g. The isotropic scattering case, $g = 0$, the pattern similarity is lost at the third column, whereas for the forward scattering case, $g = 0.9$, correlation is preserved. We verify the accuracy of our algorithm against an exact, yet computationally heavy, wave solver, as well as against analytical formulas derived under limiting assumptions.

We present a Monte Carlo rendering framework for the physically-accurate simulation of speckle patterns arising from volumetric scattering of cohere... we derive from first principles. We use this formulation to develop two Monte Carlo rendering algorithms for computing speckle covariance as well

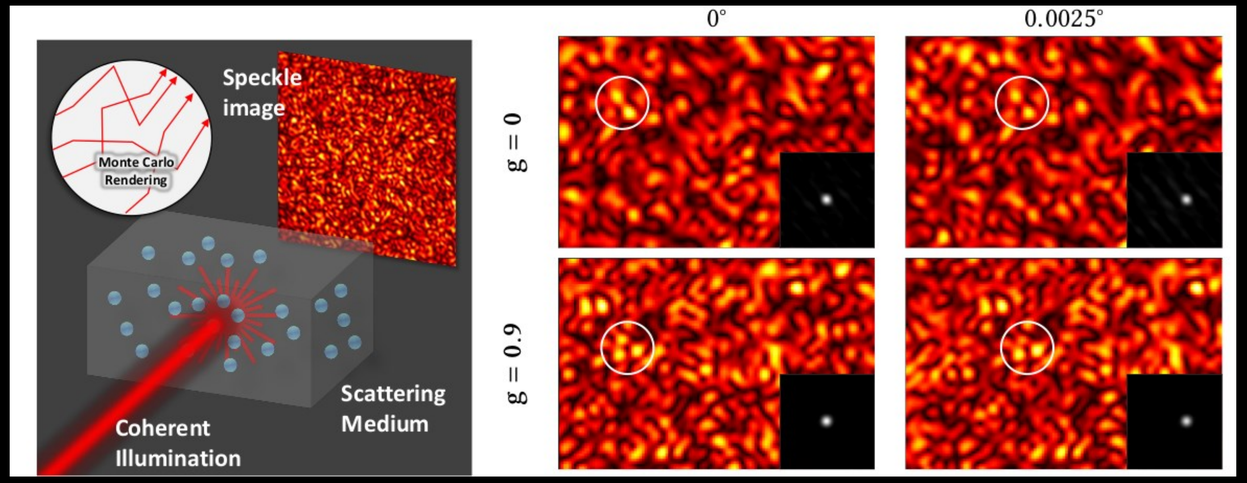
See paper for details

[Bar et al., 2019]

Comparison with wave-equation solver and real measurements



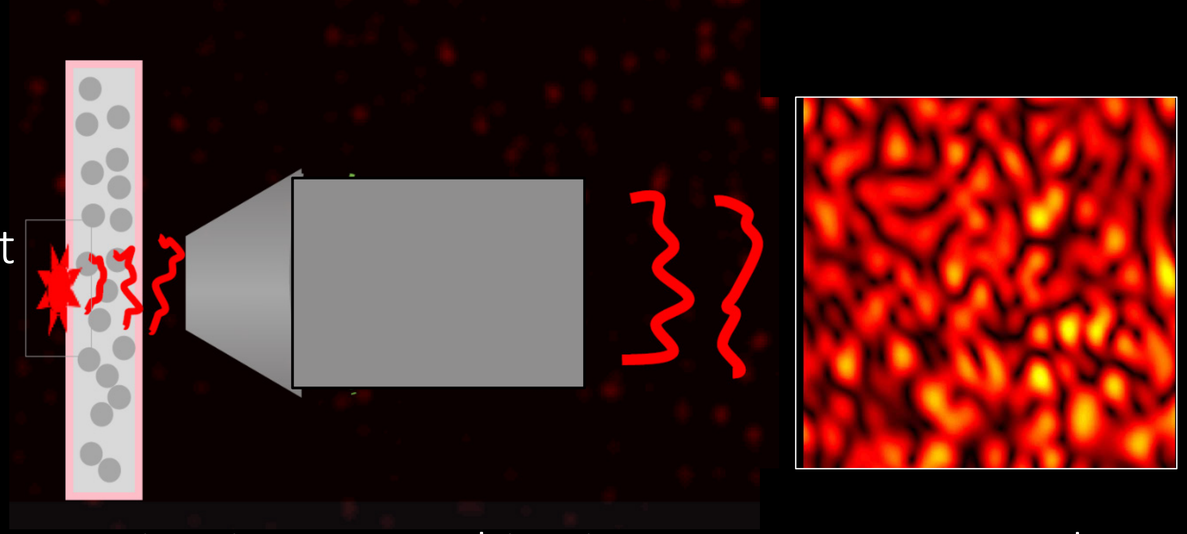
match wave equation solvers, **10^5x faster**

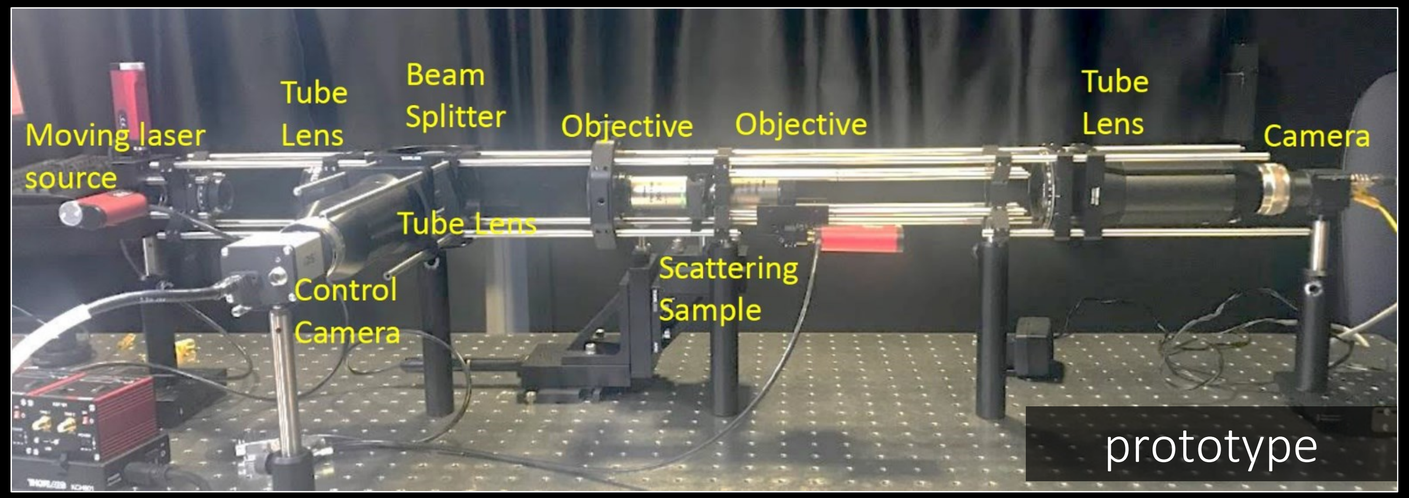


match real measurements of memory effect

[Bar et al., 2019]

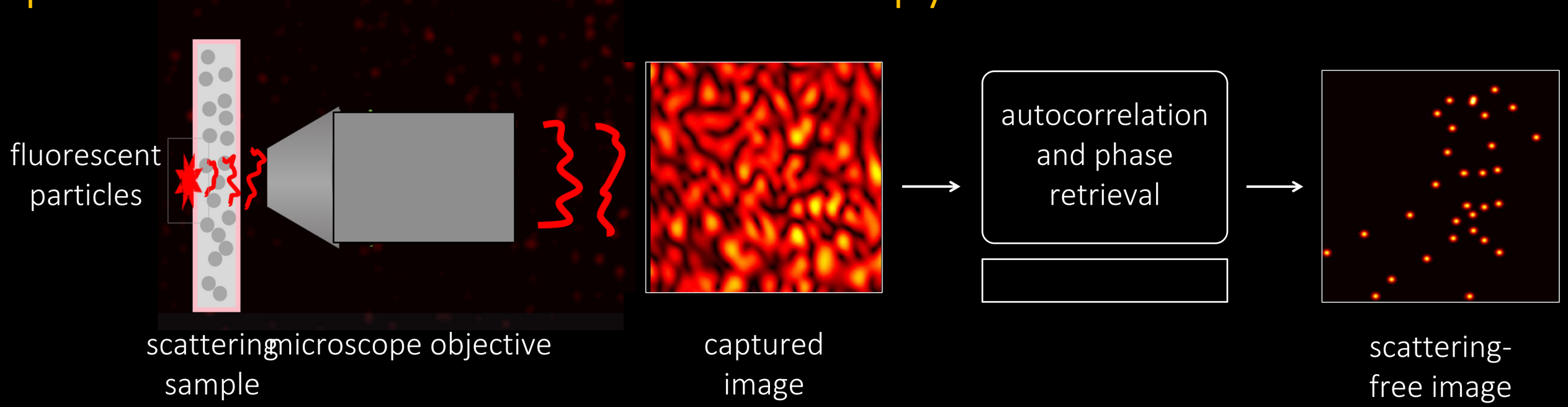
Speckle-based fluorescence microscopy

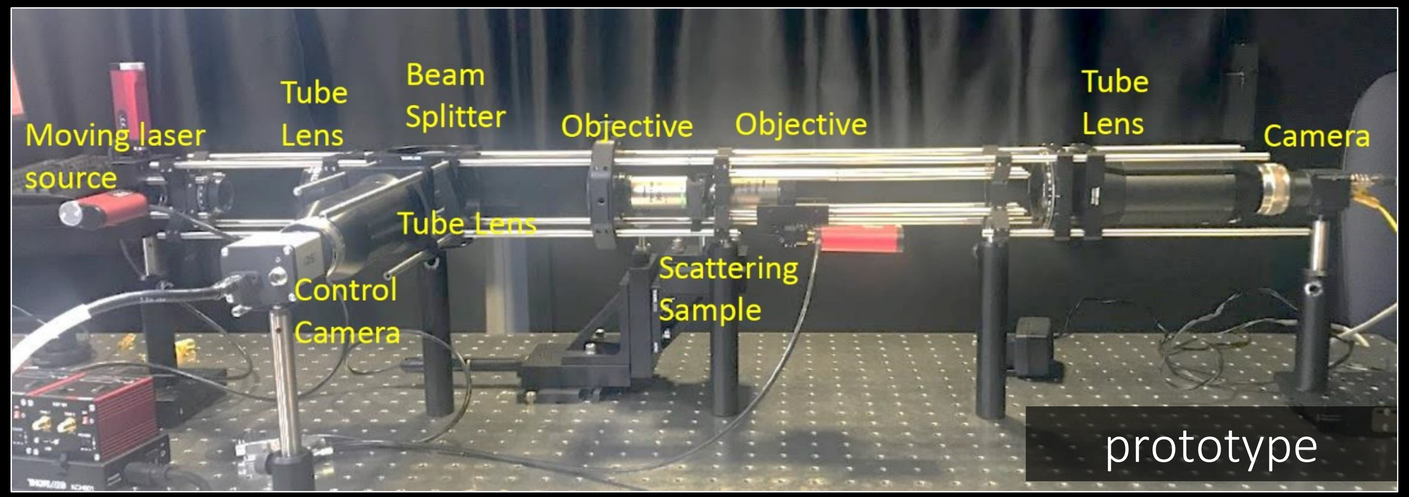




[Alterman et al., 2021]

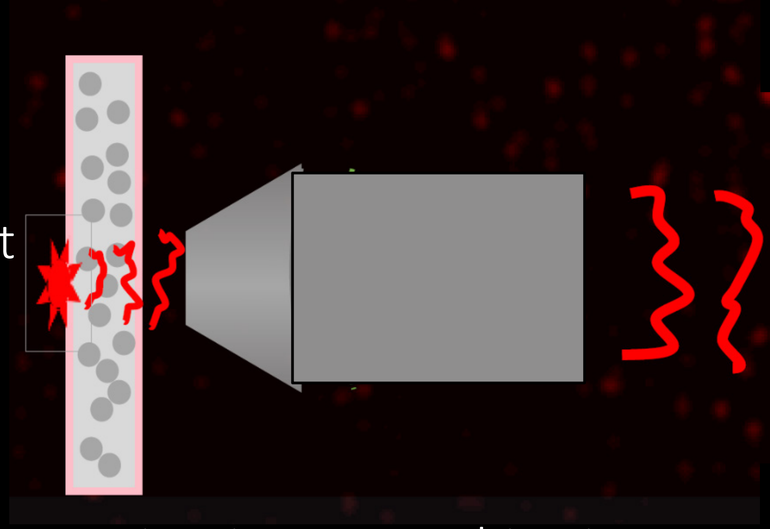
Speckle-based fluorescence microscopy

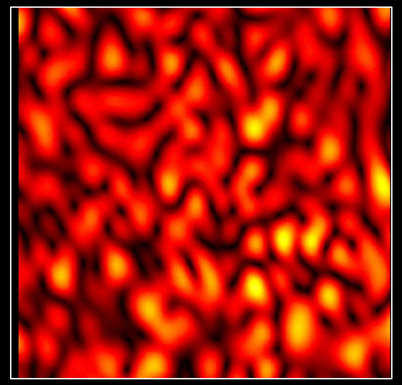




[Alterman et al., 2021]

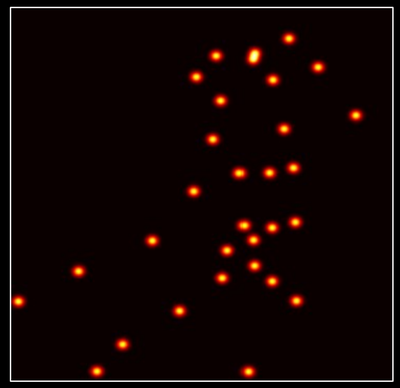
Speckle-based fluorescence microscopy



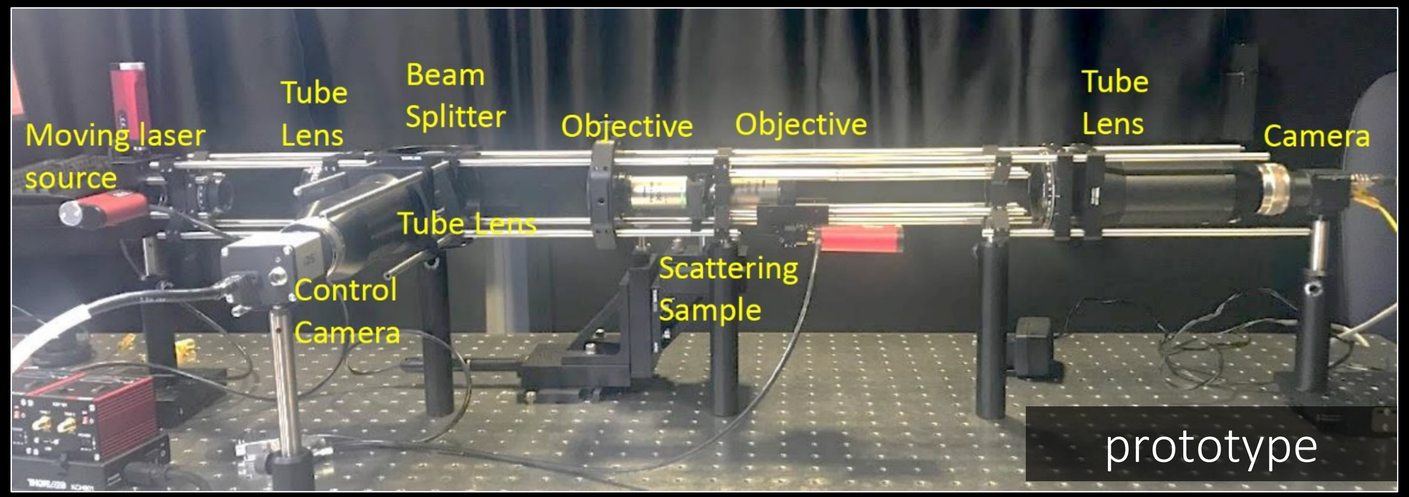


captured image

autocorrelation and phase retrieval



scattering-free image

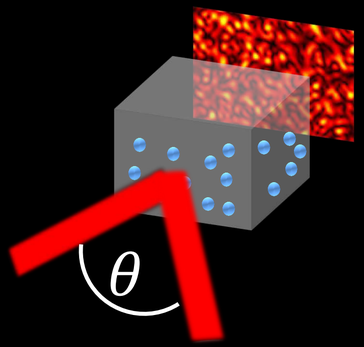


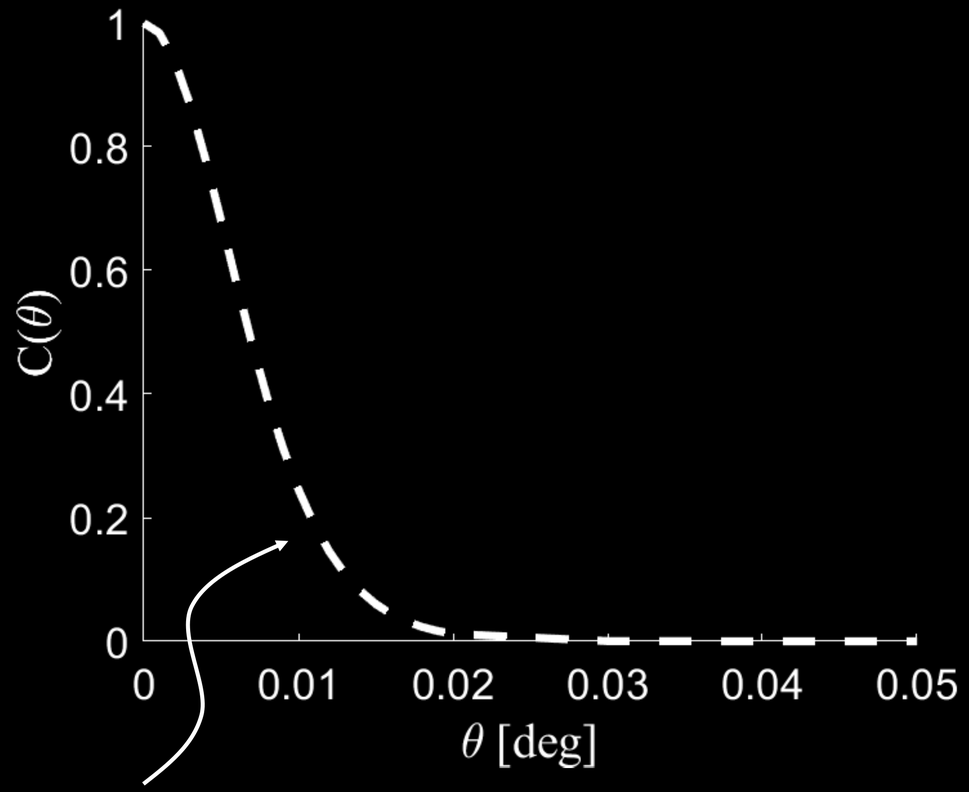
Performance strongly depends on:

- speckle statistics
- image priors
- tissue parameters

[Alterman et al., 2021]

Evaluate the memory effect

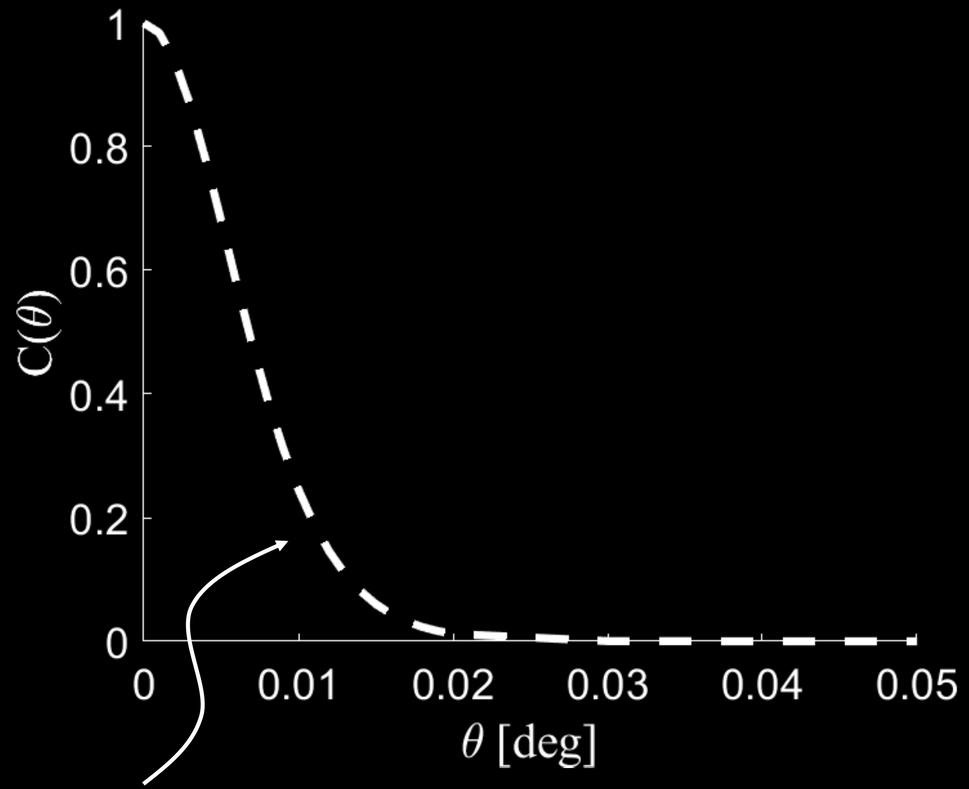




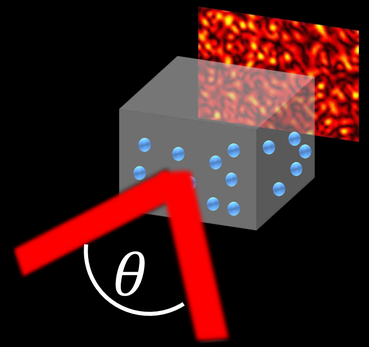
Analytical solution based on diffusion approximation

[Alterman et al., 2021]

Evaluate the memory effect



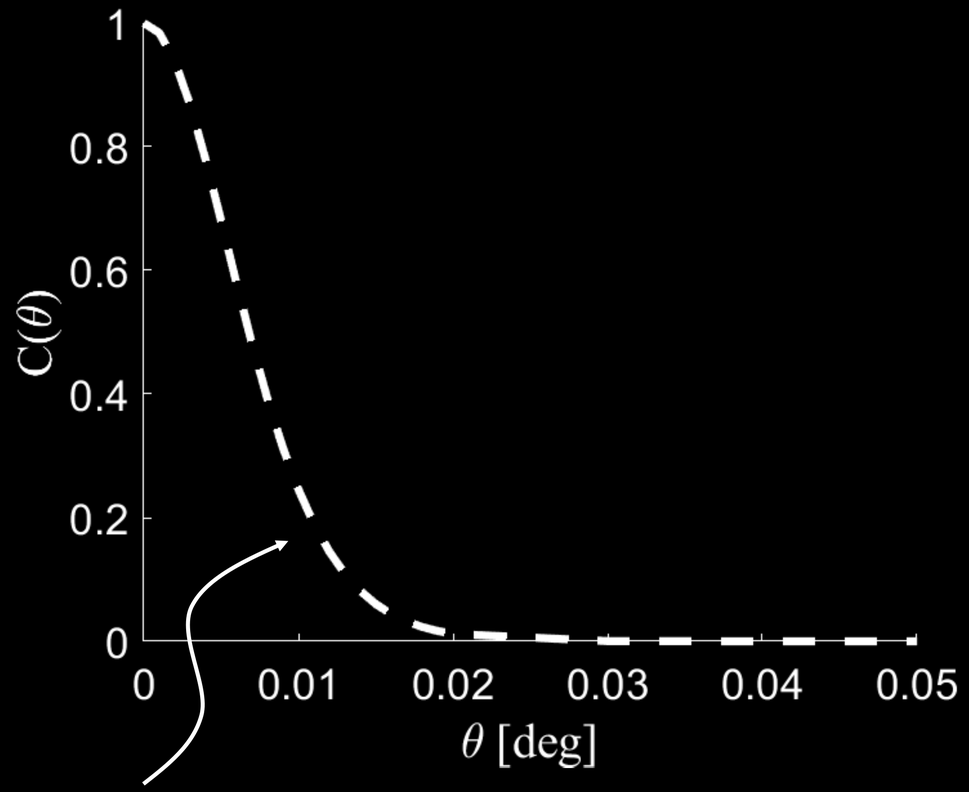
Analytical solution based on diffusion approximation



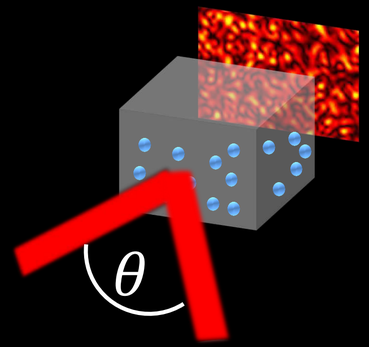
This approximation is correct for some materials and configurations

[Alterman et al., 2021]

Evaluate the memory effect



Analytical solution based on diffusion approximation

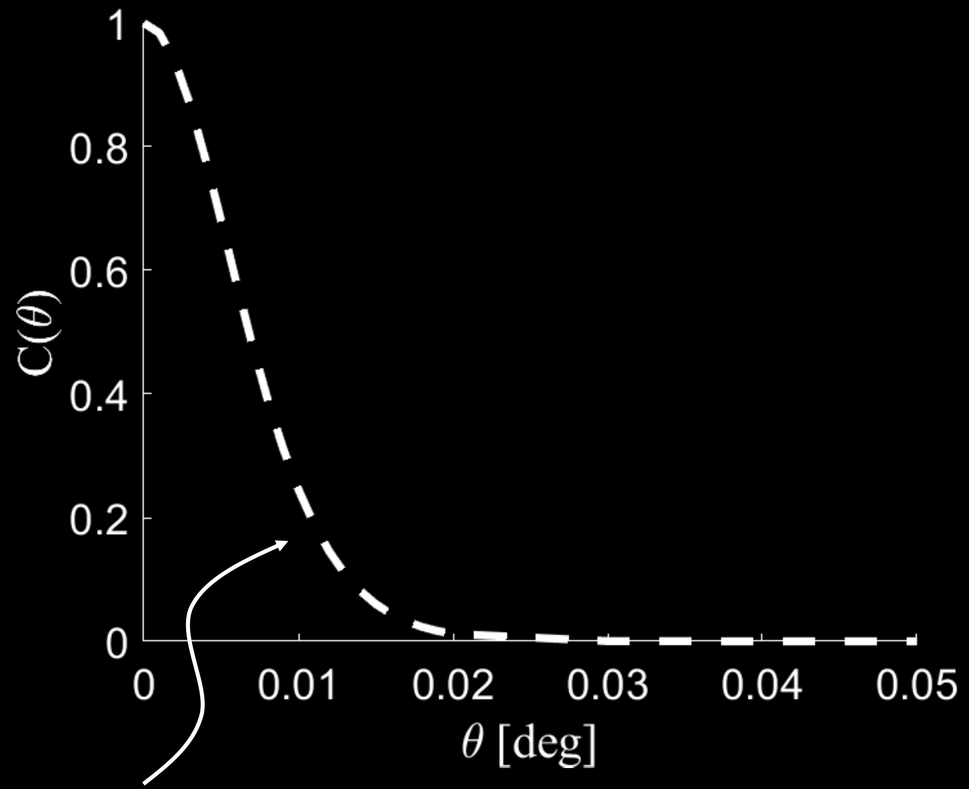


This approximation is correct for some materials and configurations

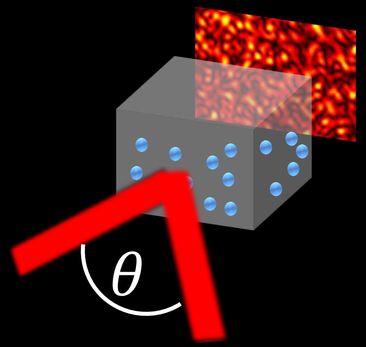
In practice ME extent is often wider

[Alterman et al., 2021]

Evaluate the memory effect



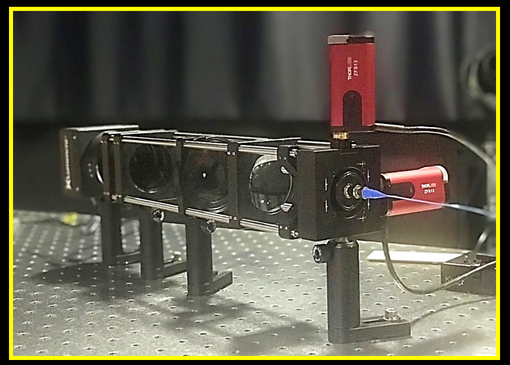
Analytical solution based on diffusion approximation



This approximation is correct for some materials and configurations

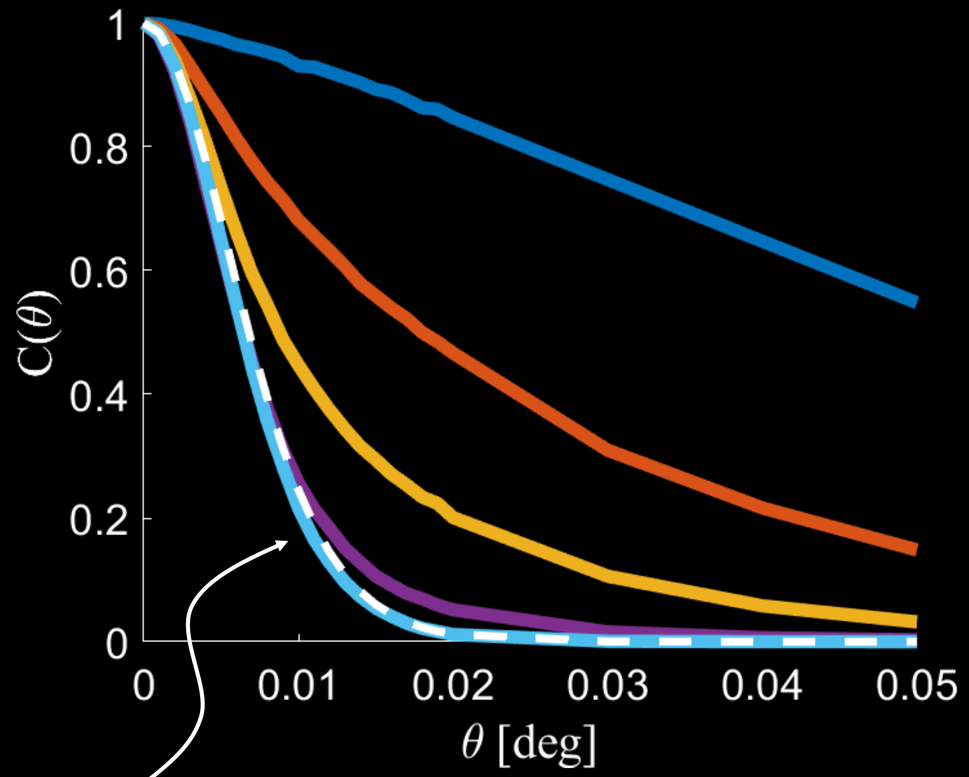
In practice ME extent is often wider

Currently: measured *empirically* in the lab

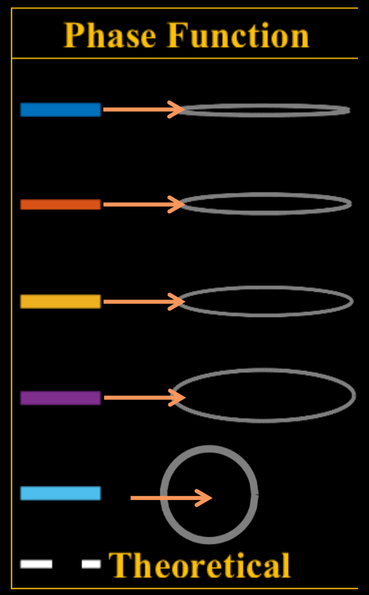


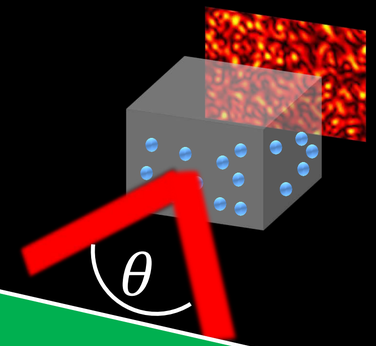
[Alterman et al., 2021]

Evaluate the memory effect



Analytical solution based on diffusion approximation



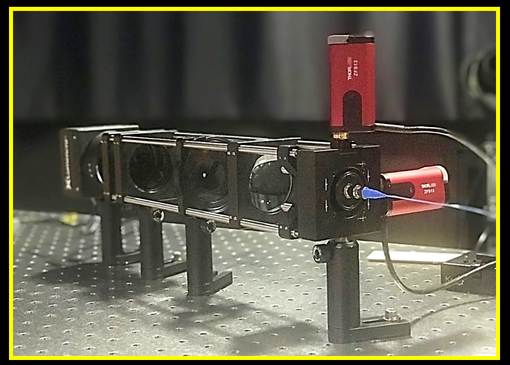


This ar… …e

In practice ME exte… …r

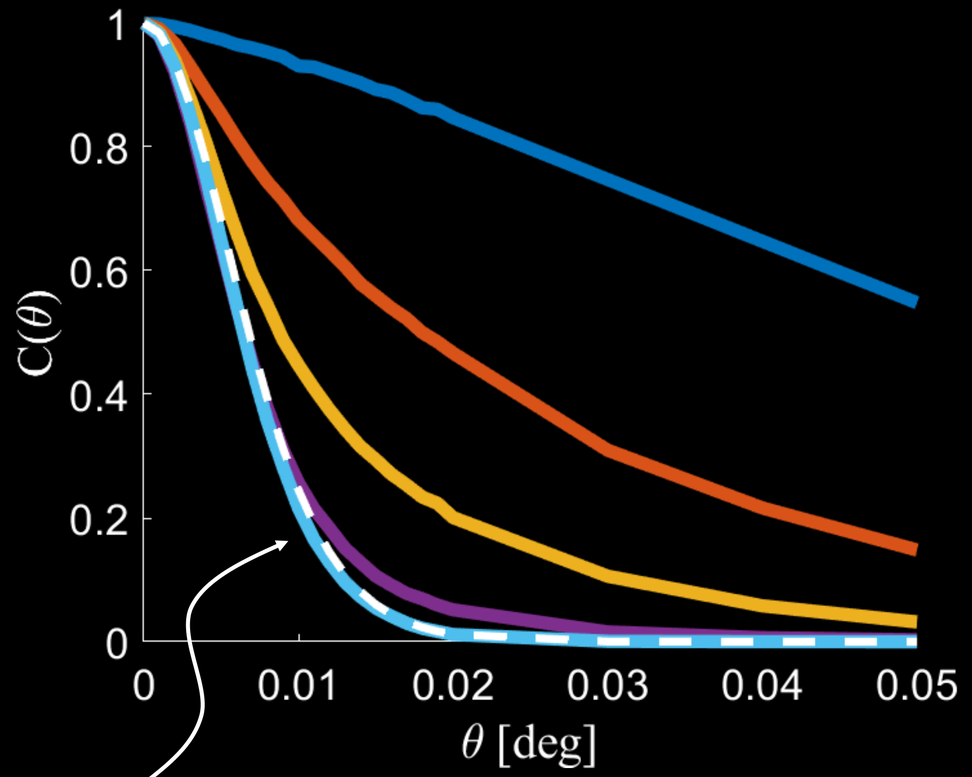
Now we can efficiently compute ME curves for all scattering parameters

Currently: measured *empirically* in the lab

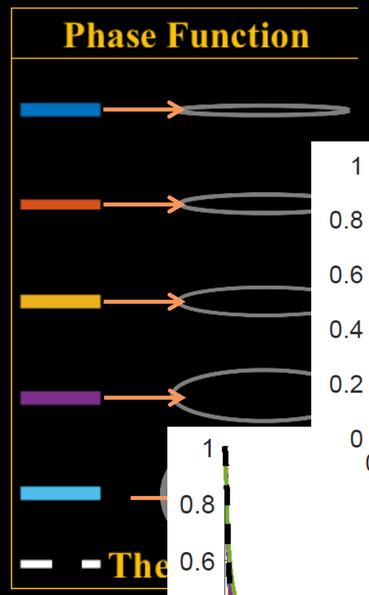


[Alterman et al., 2021]

Evaluate the memory effect



Analytical solution based on diffusion approximation

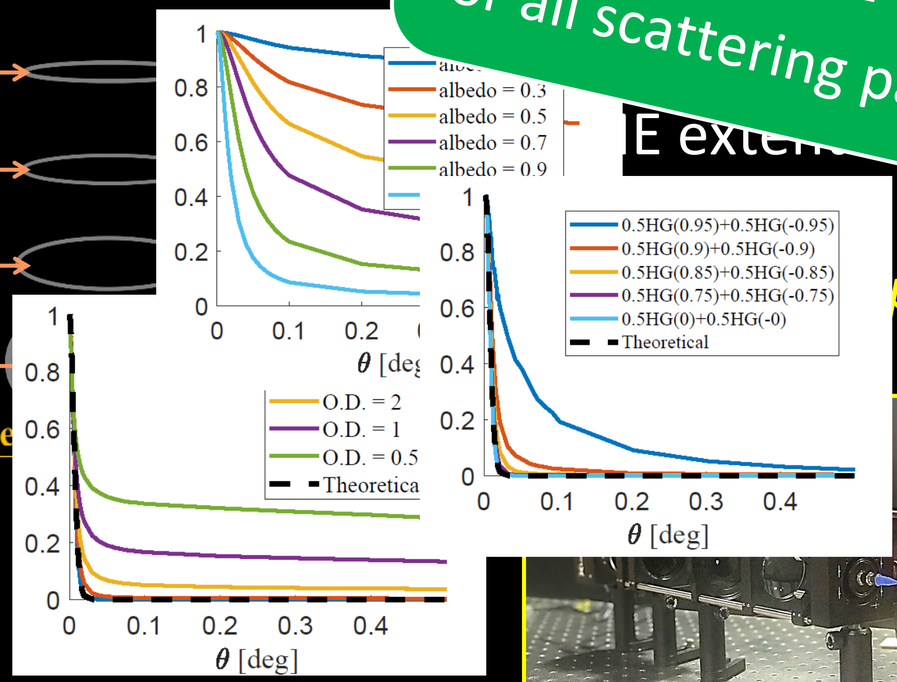


This ar e

E exten r

pirically in the

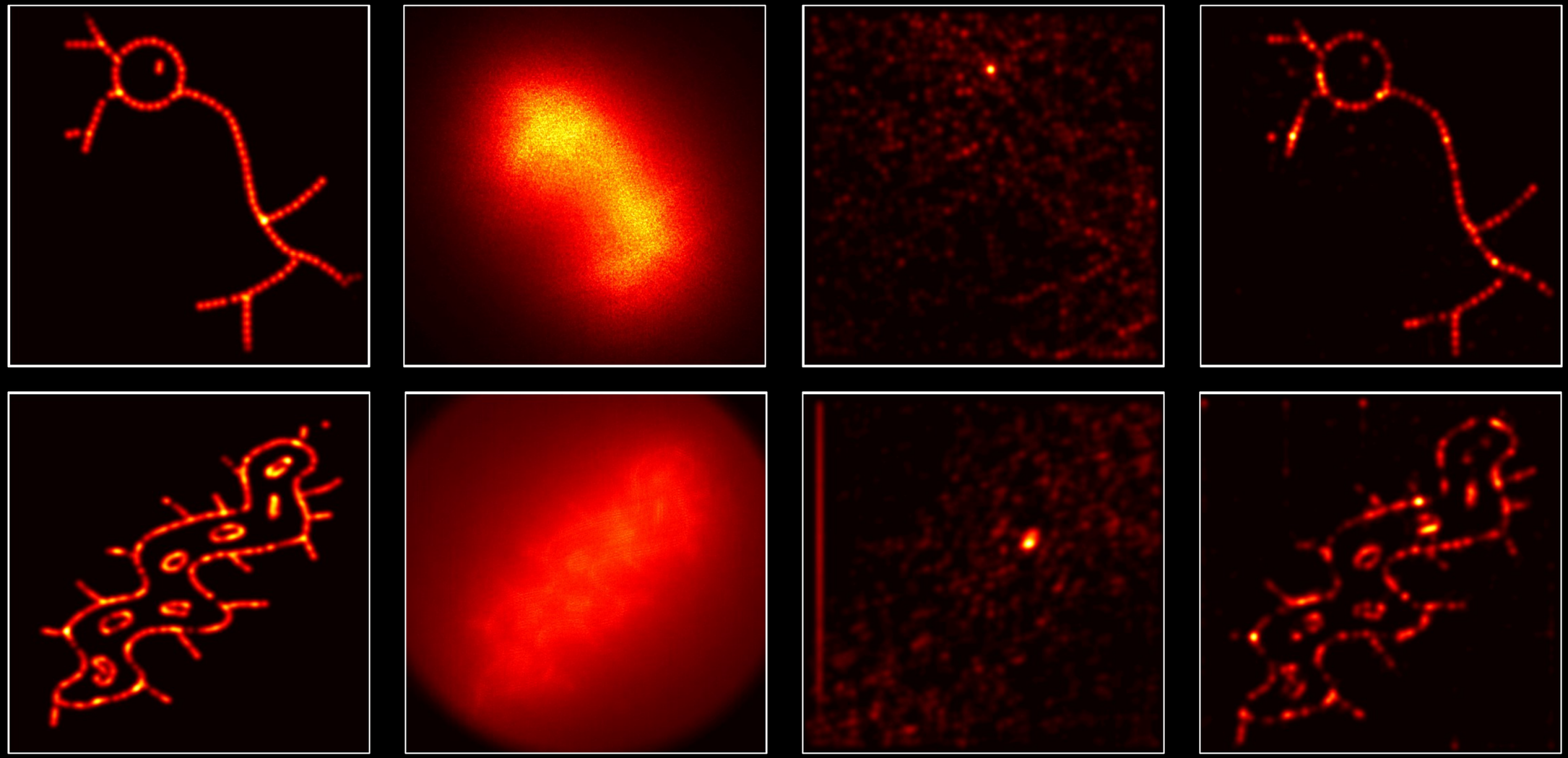
Now we can efficiently compute ME curves for all scattering parameters



[Alterman et al., 2021]

Better algorithms for fluorescence microscopy

groundtruth | input image | prior algorithm | our algorithm



[Alterman et al., 2021]

Acquisition of scattering materials

Use differentiable speckle rendering to recover material parameters from speckle images

acquisition camera
- records speckle correlations

material samples

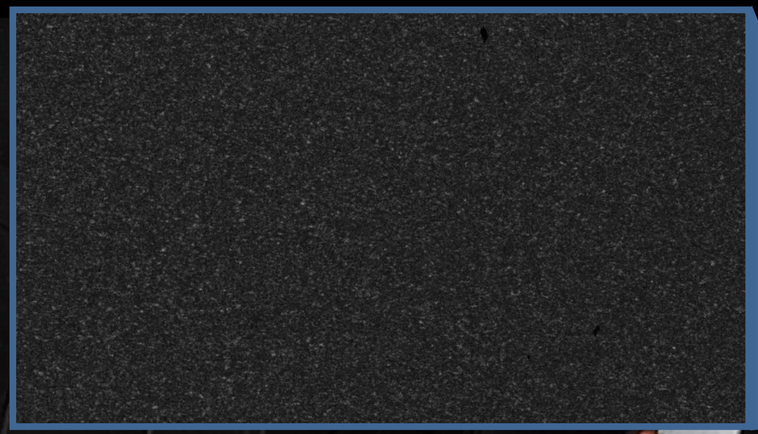
motorized sample mount
- 8 degrees of freedom

rotating illuminator
- two laser beams at 4° separation

high-power coherent monochromatic laser

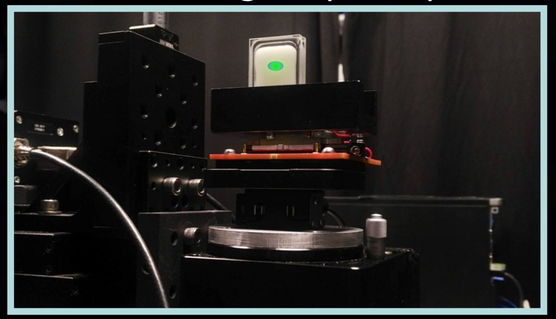
Optical fiber

Acquisition



Speckle Video
(Camera Feed)

Scattering Soap Sample



Rendering wave optics is a very active area

A Generic Framework for Physical Light Transport

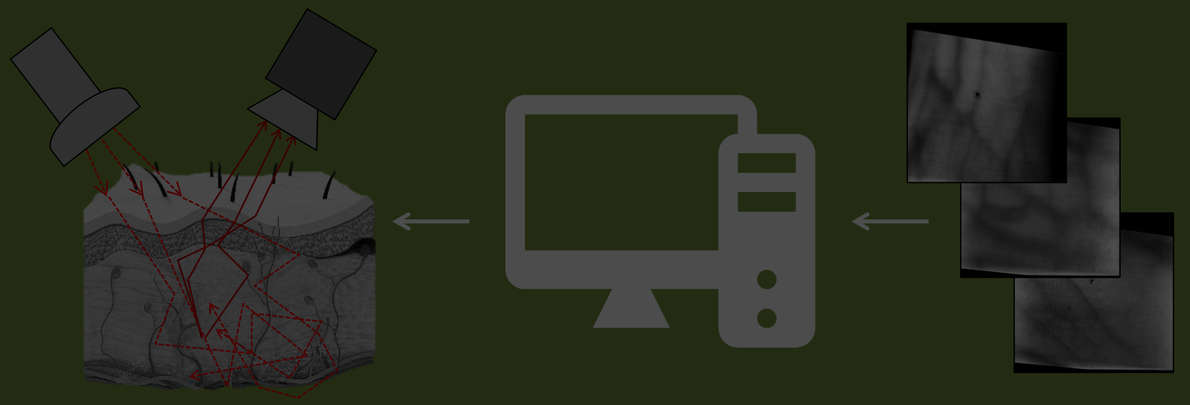
SHLOMI STEINBERG, University of California, Santa Barbara, USA
LING-QI YAN, University of California, Santa Barbara, USA

Physics-based rendering and its applications to computational imaging

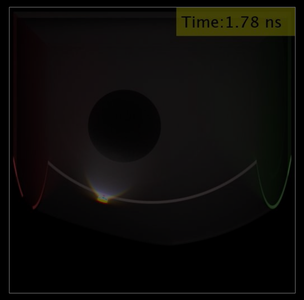
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

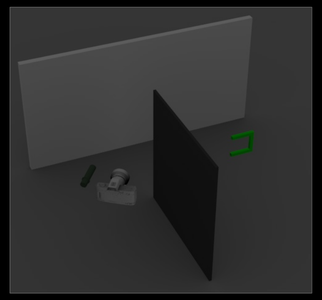
inverse rendering



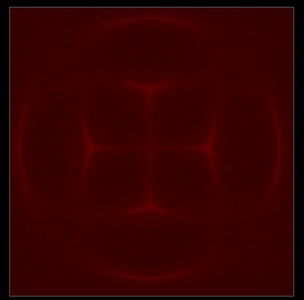
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



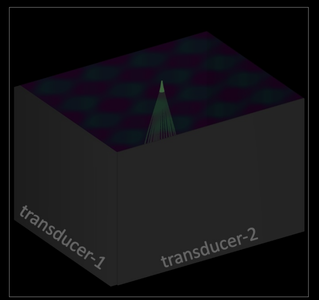
time-of-flight imaging



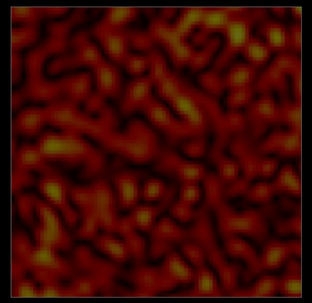
non-line-of-sight imaging



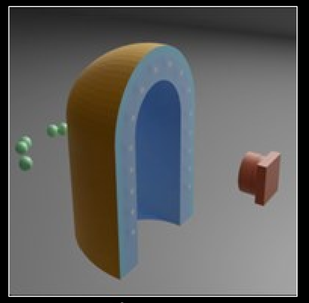
acousto-optic lensing



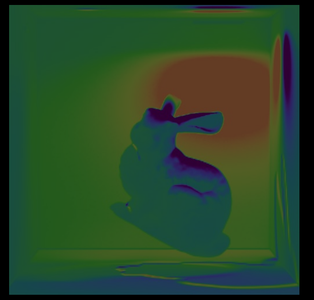
ultrafast light scanning



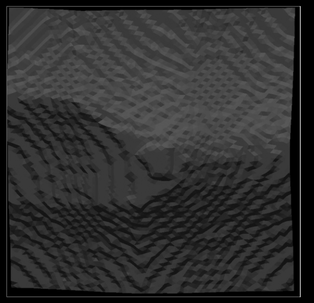
speckle imaging



tactile sensor design

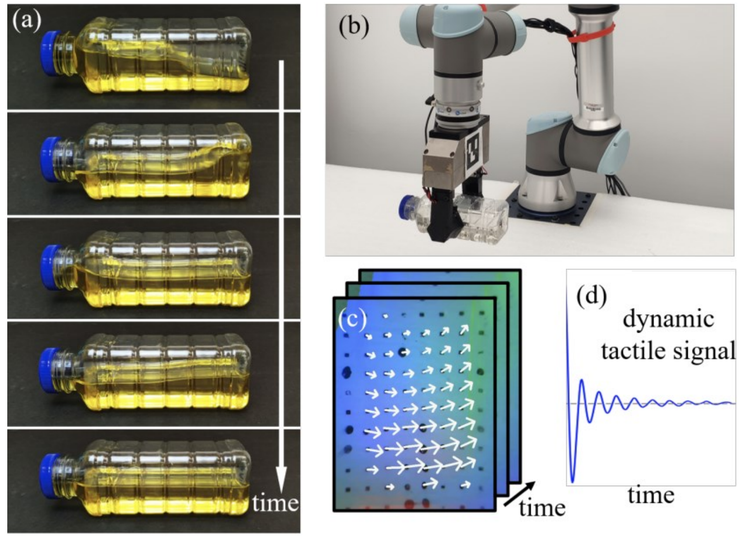


differentiable renderer

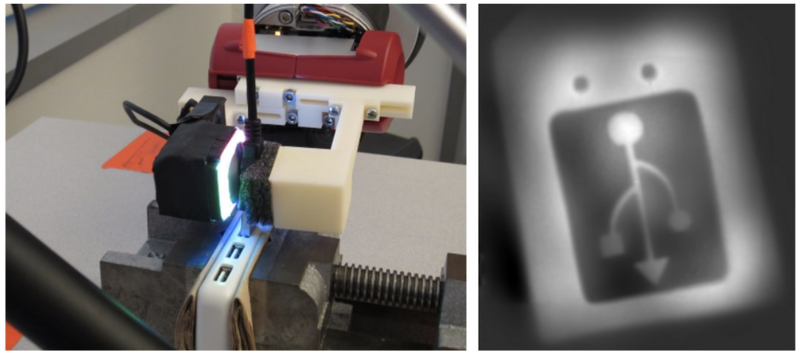


inverse problems

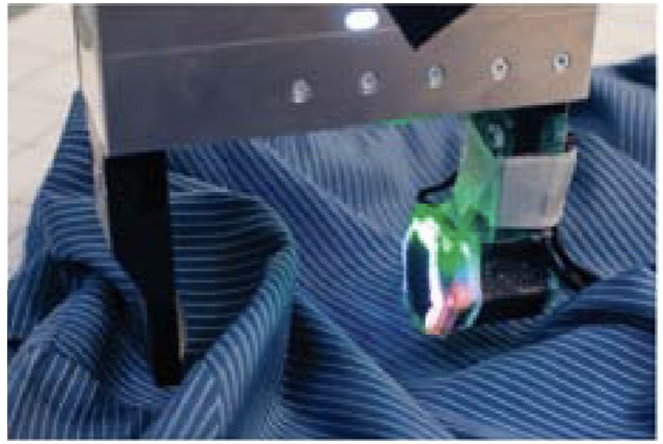
Why tactile sensing?

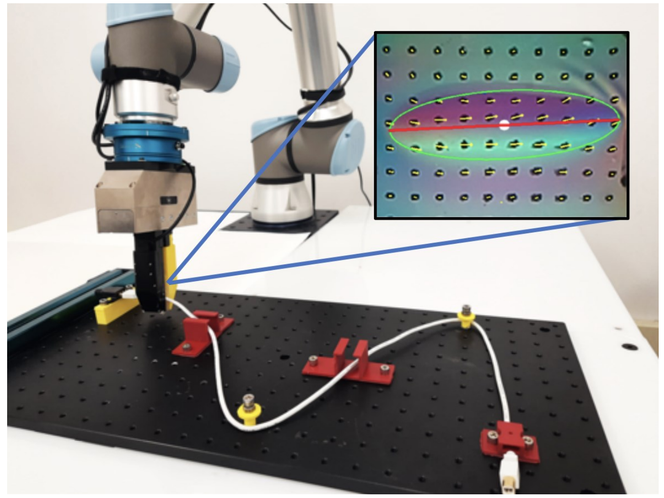


Object perception
[Huang et.al. 2022]

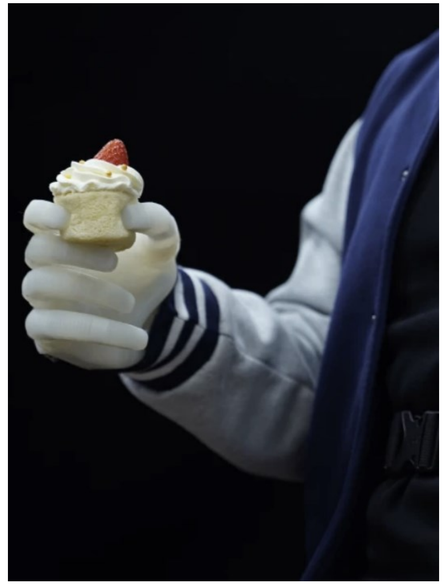


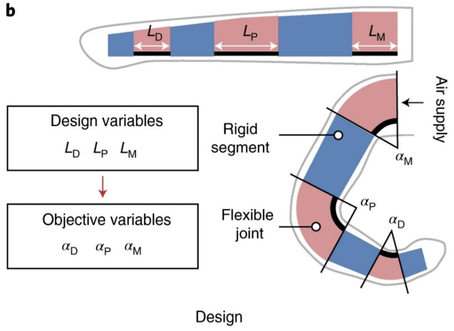
Advanced manufacturing
[Li et.al. 2014]

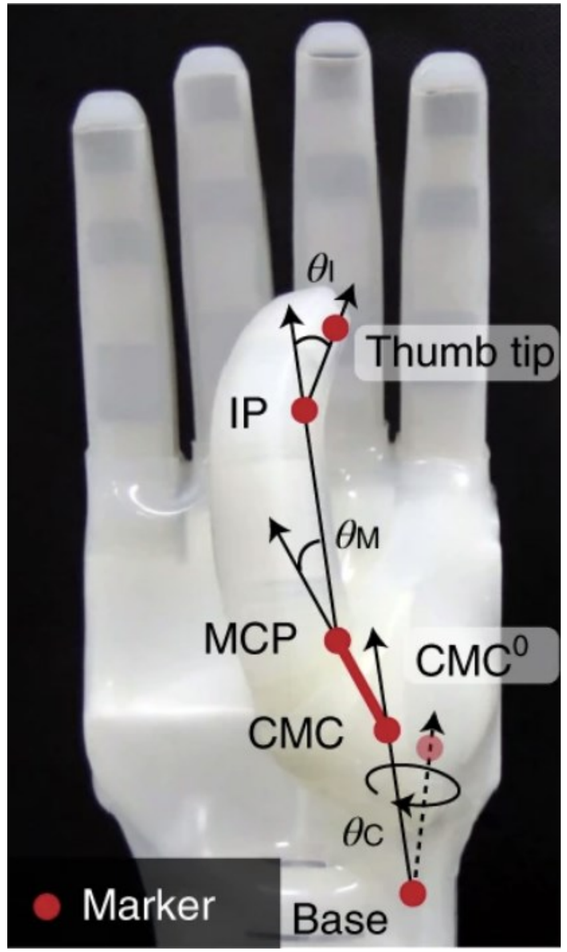




Robotic manipulation
[Yuan et.al. 2018]
[Wilson et.al. 2023]



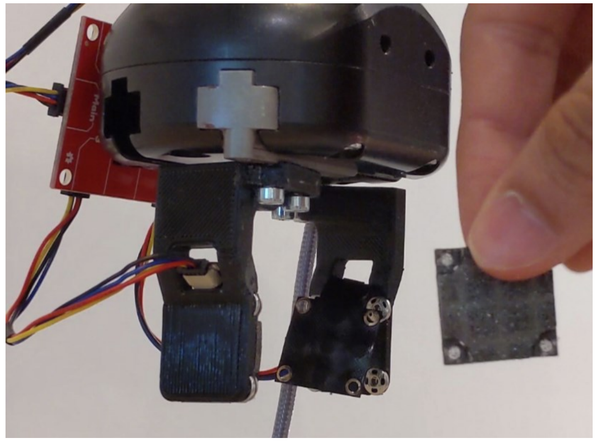




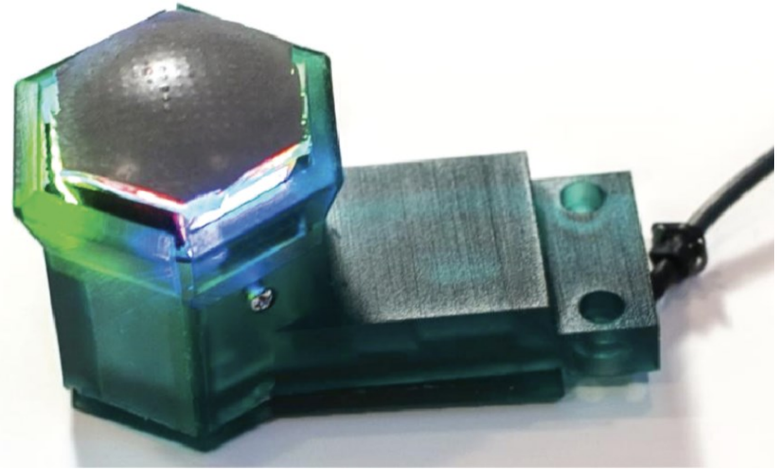
Neuroprosthetics
[Gu et.al. 2023]

Tactile sensors

BioTac
ionically-conductive fluid based



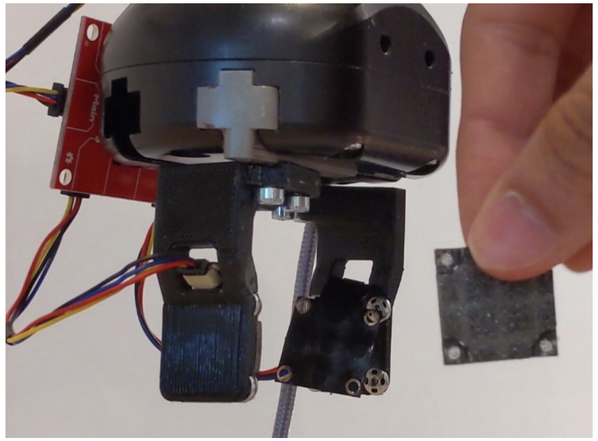
ReSkin
Magnetic field based



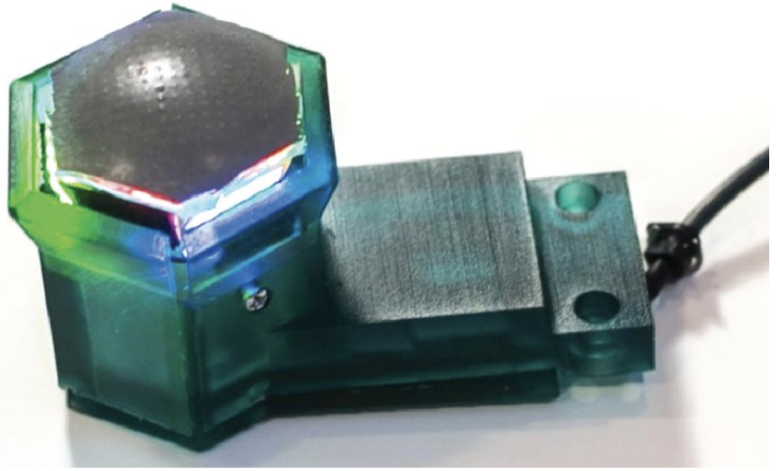
GelSight
Vision-based

Tactile sensors

BioTac
ionically-conductive fluid based

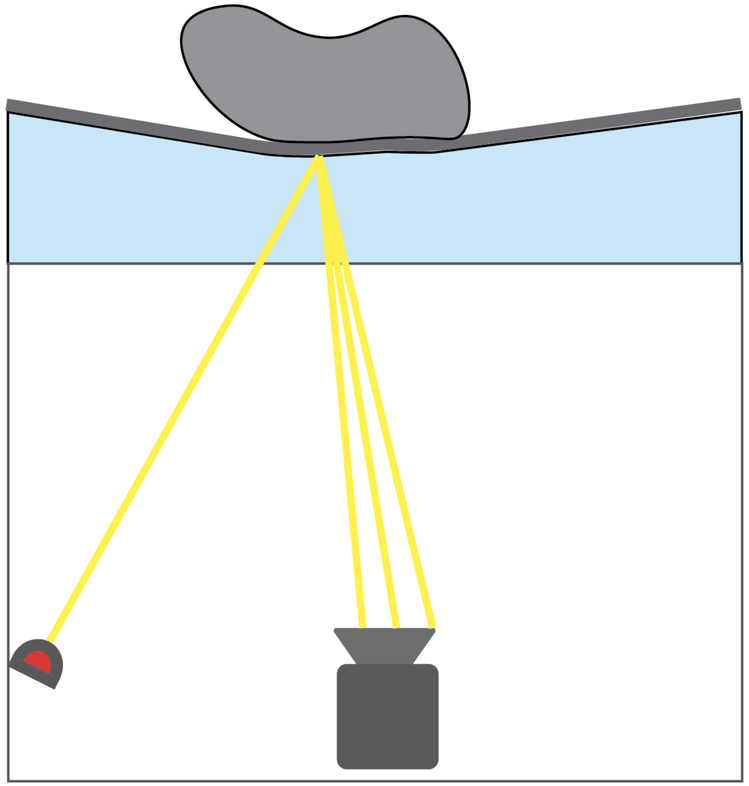


ReSkin
Magnetic field based

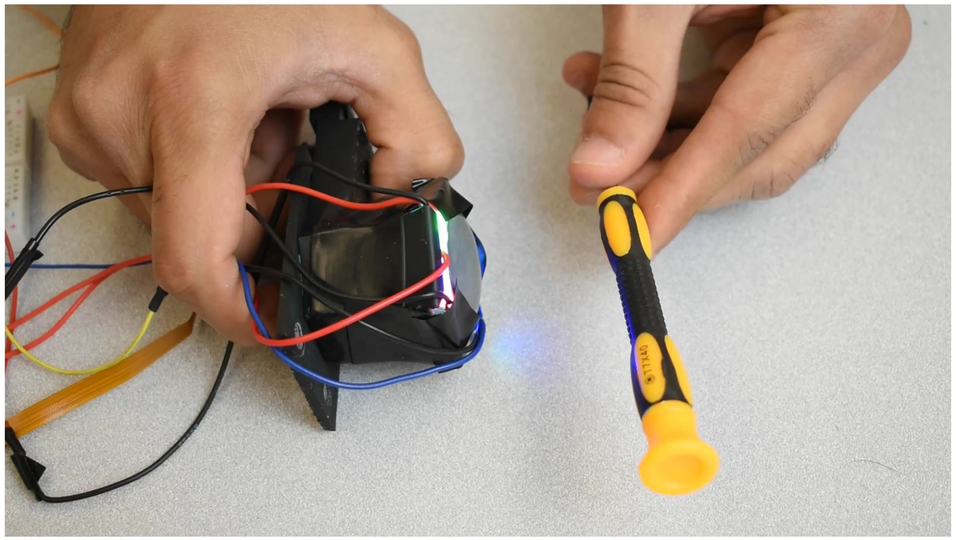


GelSight
Vision-based

Vision-based tactile sensors: working principle



Sensor schematic

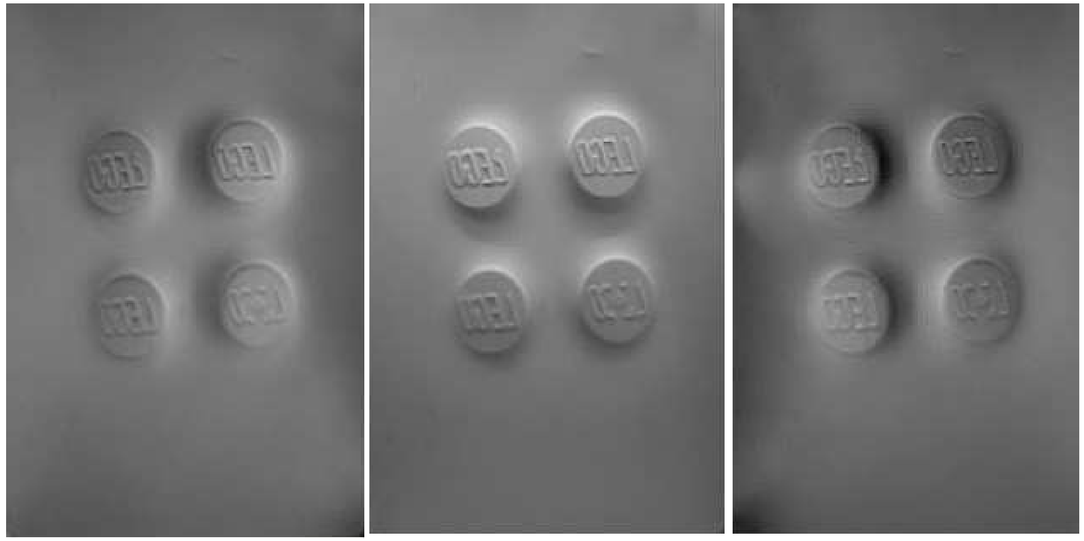


Live view

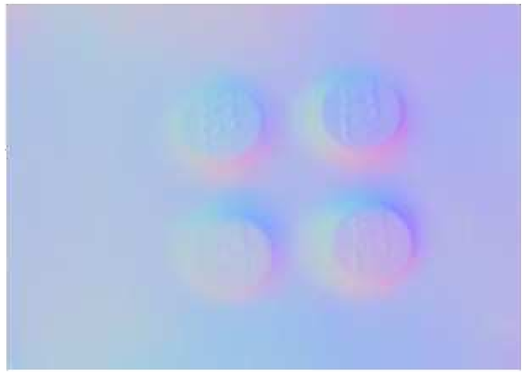
Sensor view

[Johnson and Adelson, 2021]

Vision-based tactile sensors: photometric stereo



Camera Images

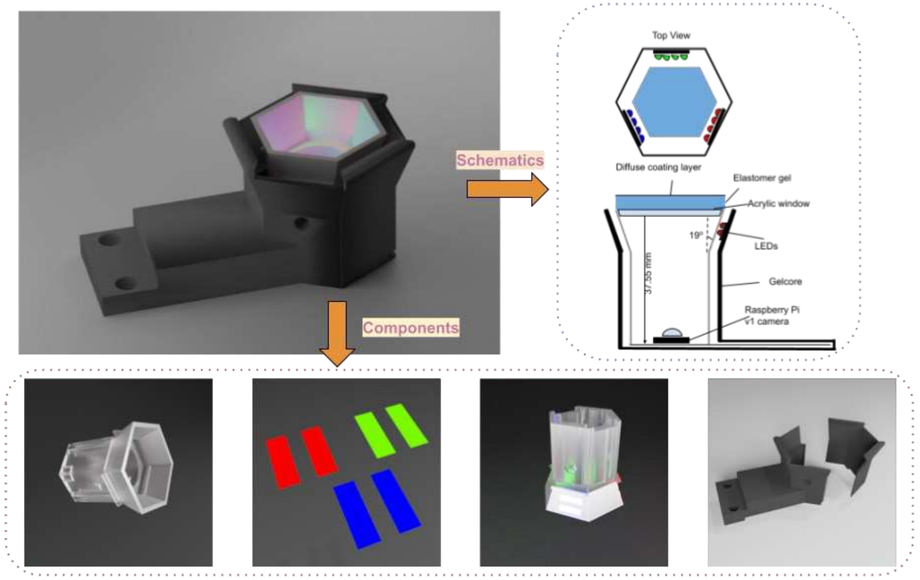


Normals

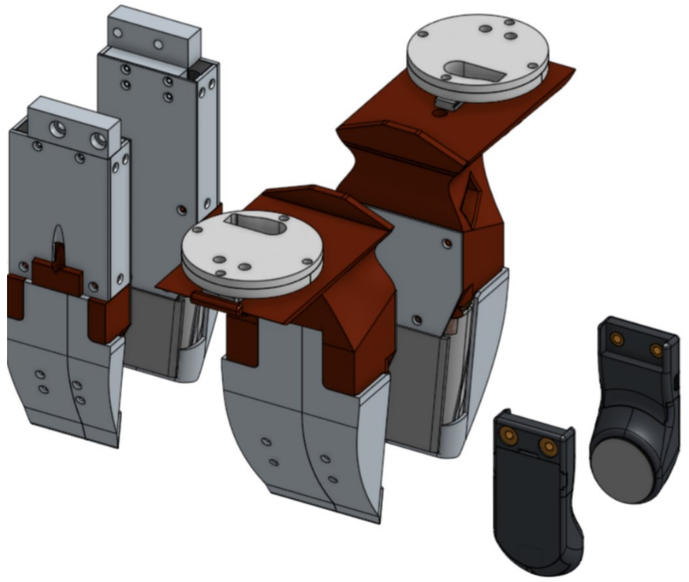
With photometric stereo, GelSight can encode surface normals as an RGB image.

[Johnson and Adelson, 2021]

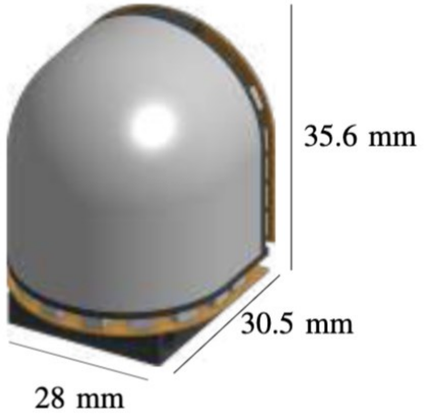
Vision-based tactile sensors: design variants



FlatGel GelSight
Dong et. al. 2017
Agarwal et. al. 2021



GelSlim Family
Donlon et. al. 2018
Ma et. al. 2019
Hogan et. al. 2020
Taylor et. al. 2021



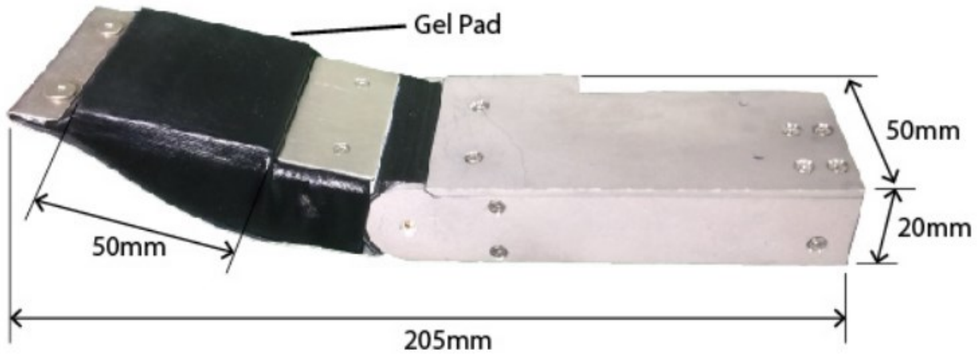
RoundTip GelSight
Romero et. al. 2020

Designing VBTS is hard

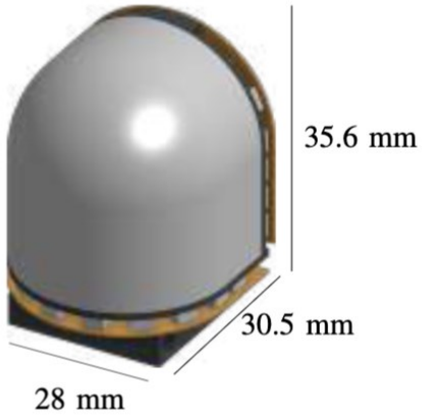
[Agarwal et al., 2023]

Designing VBTS is hard

- Diversity of sensor shape and required form-factor



Flat sensing surface

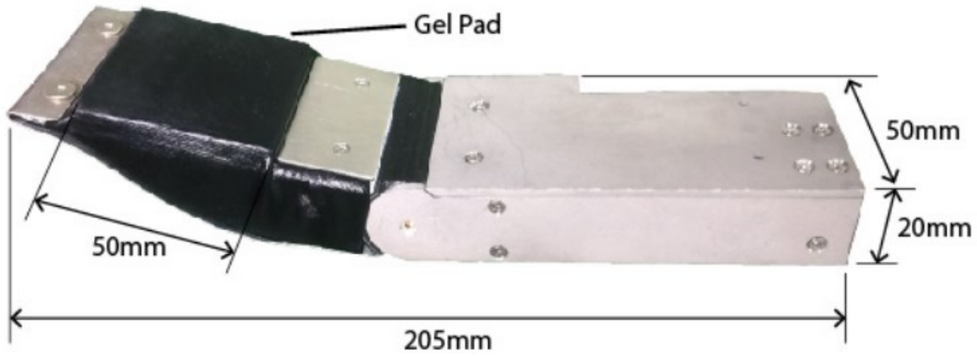


Curved sensing surface

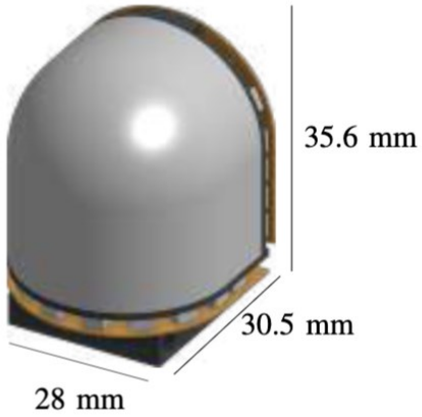
[Agarwal et al., 2023]

Designing VBTS is hard

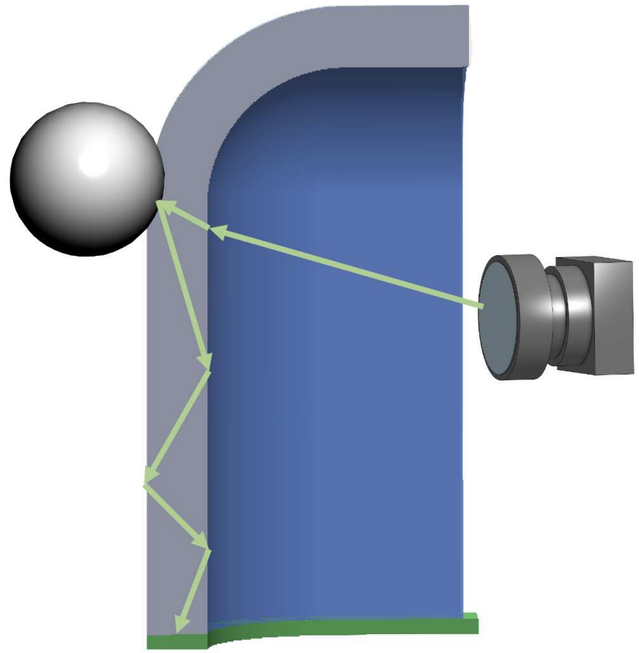
- Diversity of sensor shape and required form-factor
- Complex light interaction



Flat sensing surface



Curved sensing surface

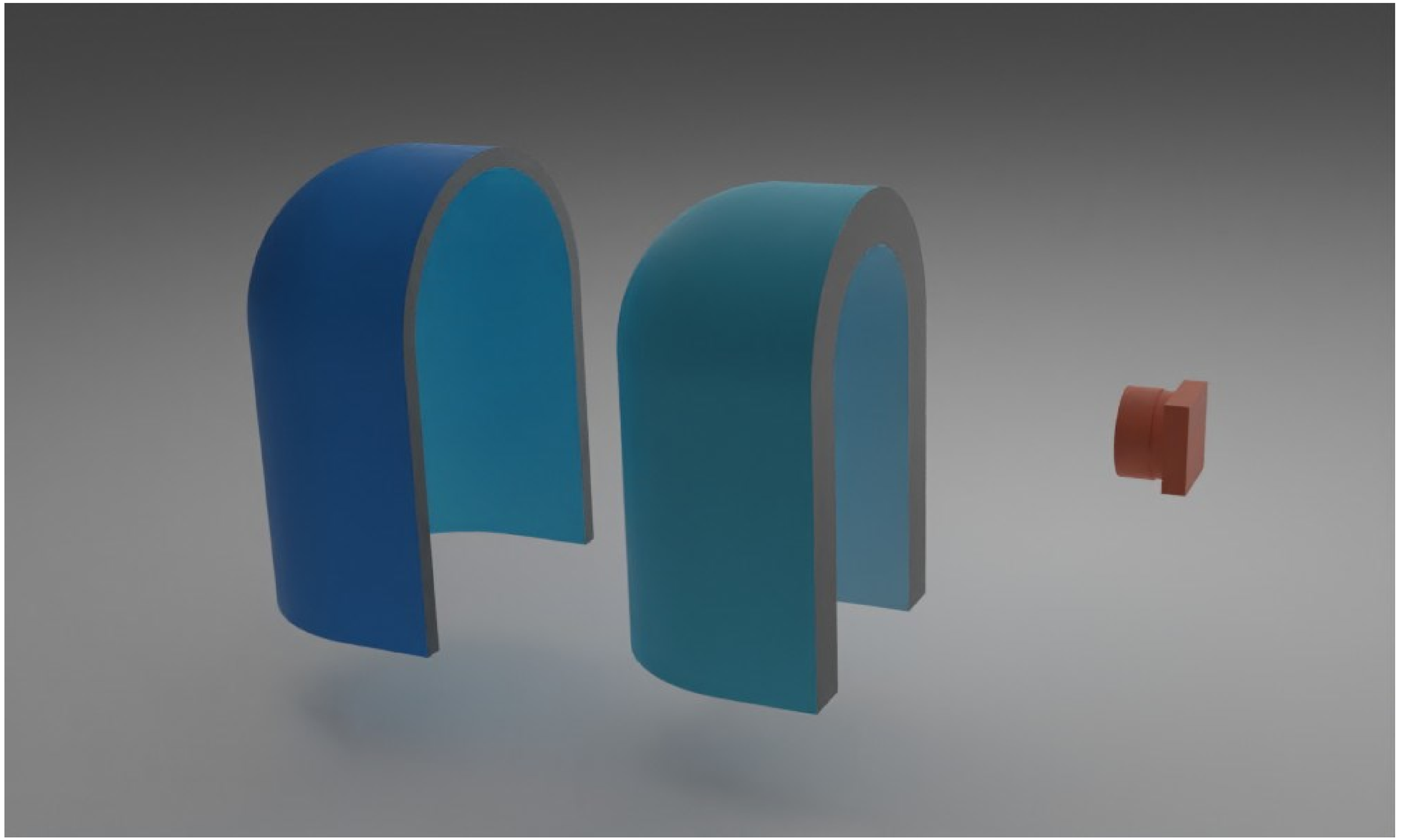


[Agarwal et al., 2023]

Curved tactile sensor

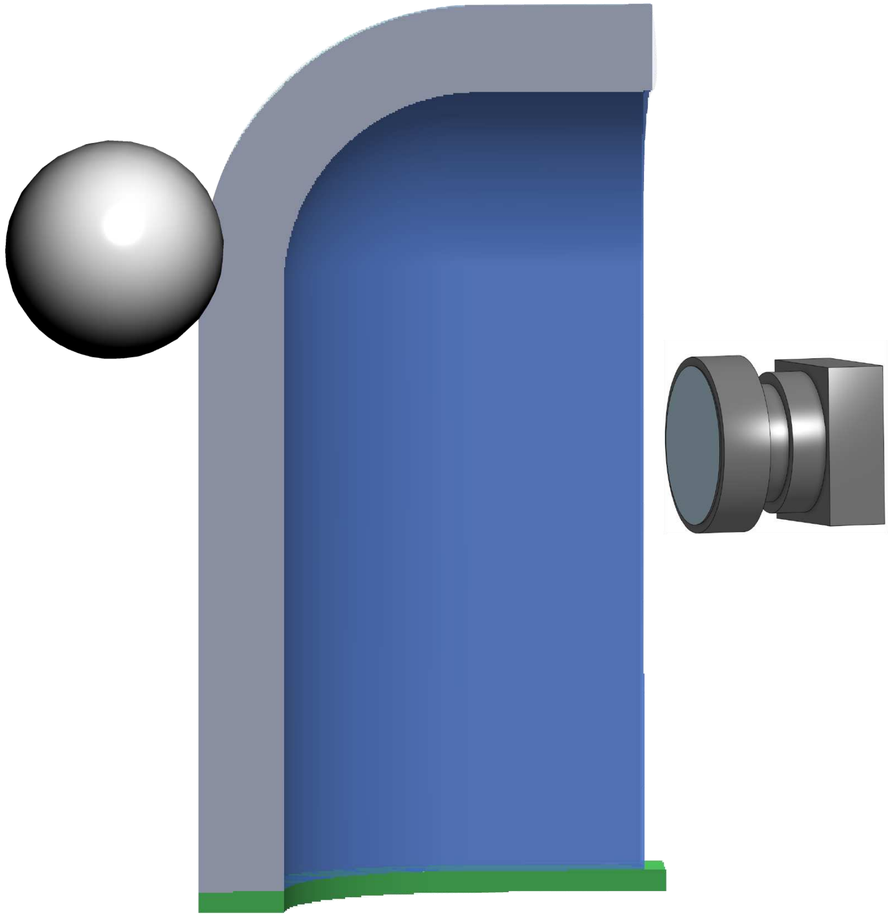
[Agarwal et al., 2023]

Curved tactile sensor

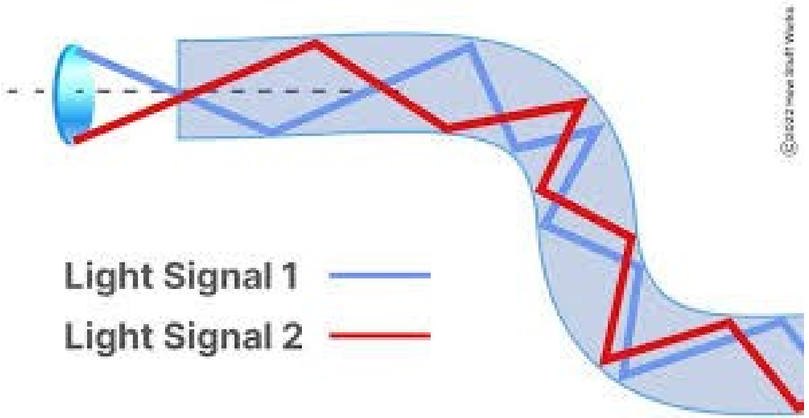


[Agarwal et al., 2023]

Curved tactile sensor: *working principle*



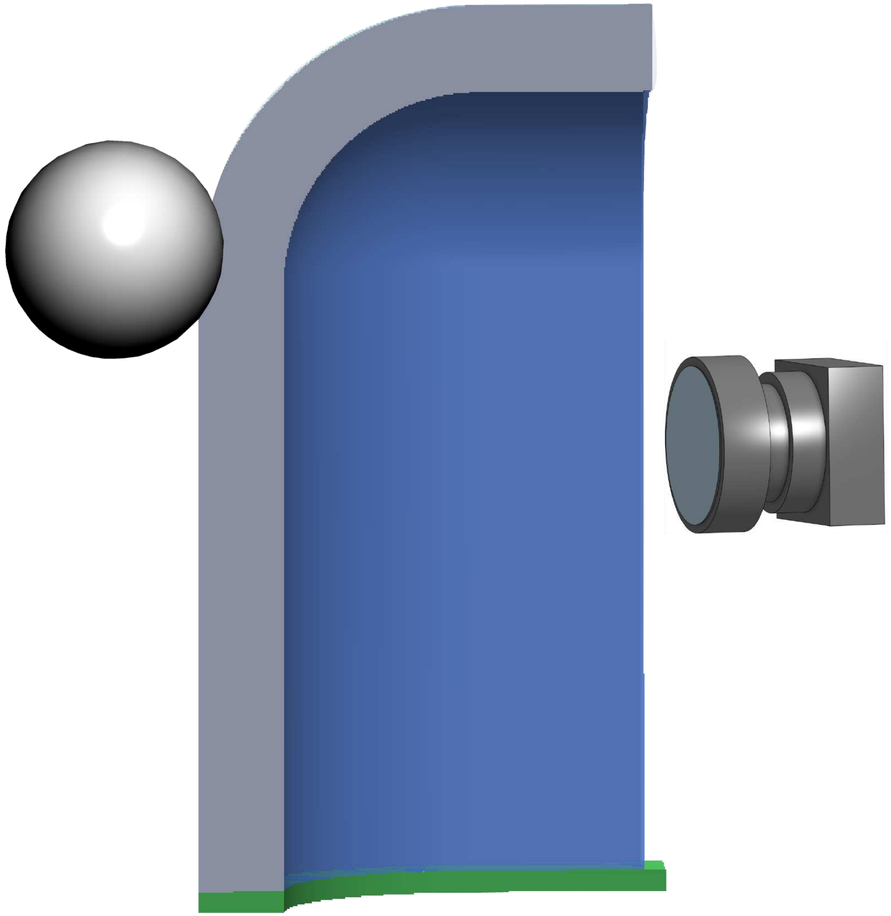
Section View



Light Piping

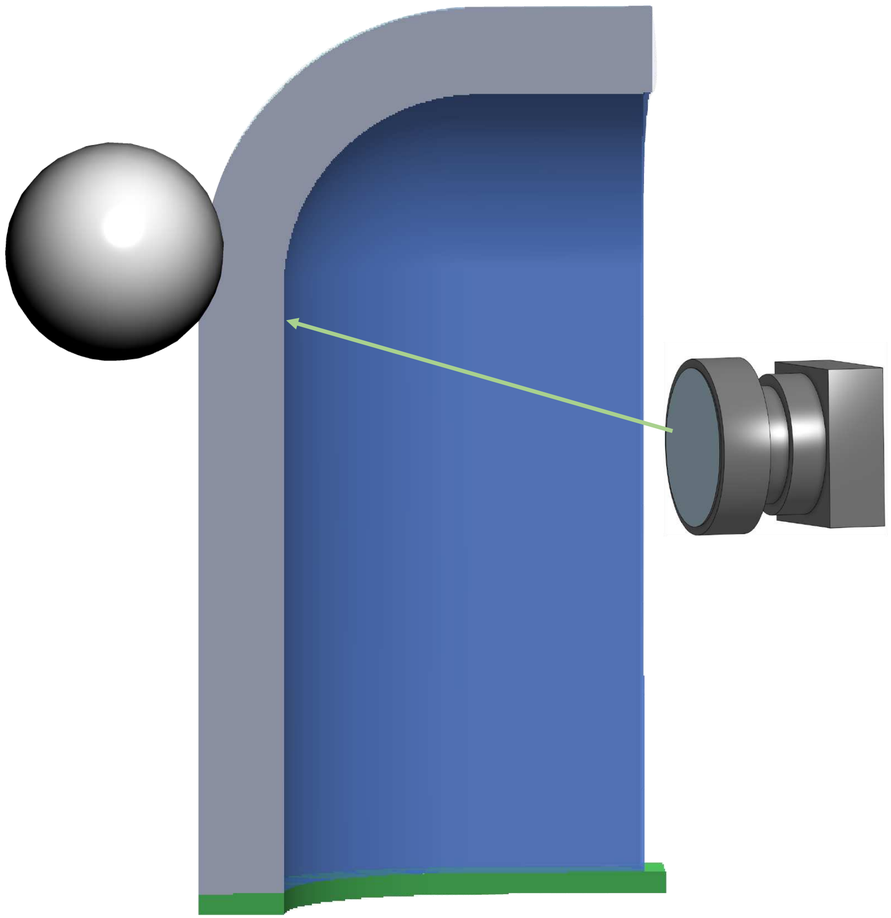
[Agarwal et al., 2023]

Curved tactile sensor: *working principle*

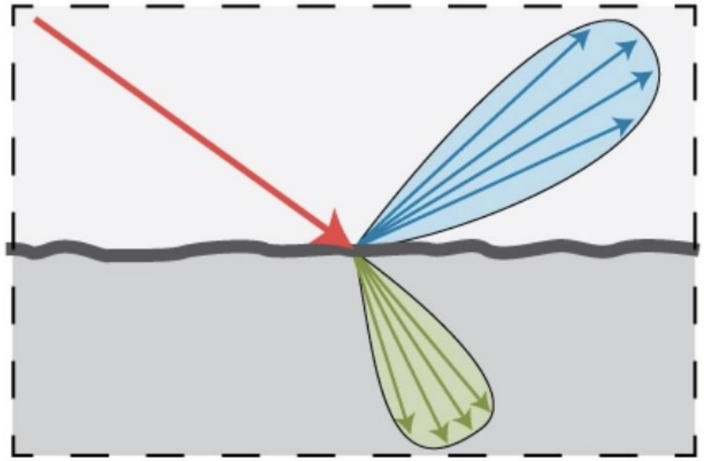


[Agarwal et al., 2023]

Curved tactile sensor: *working principle*



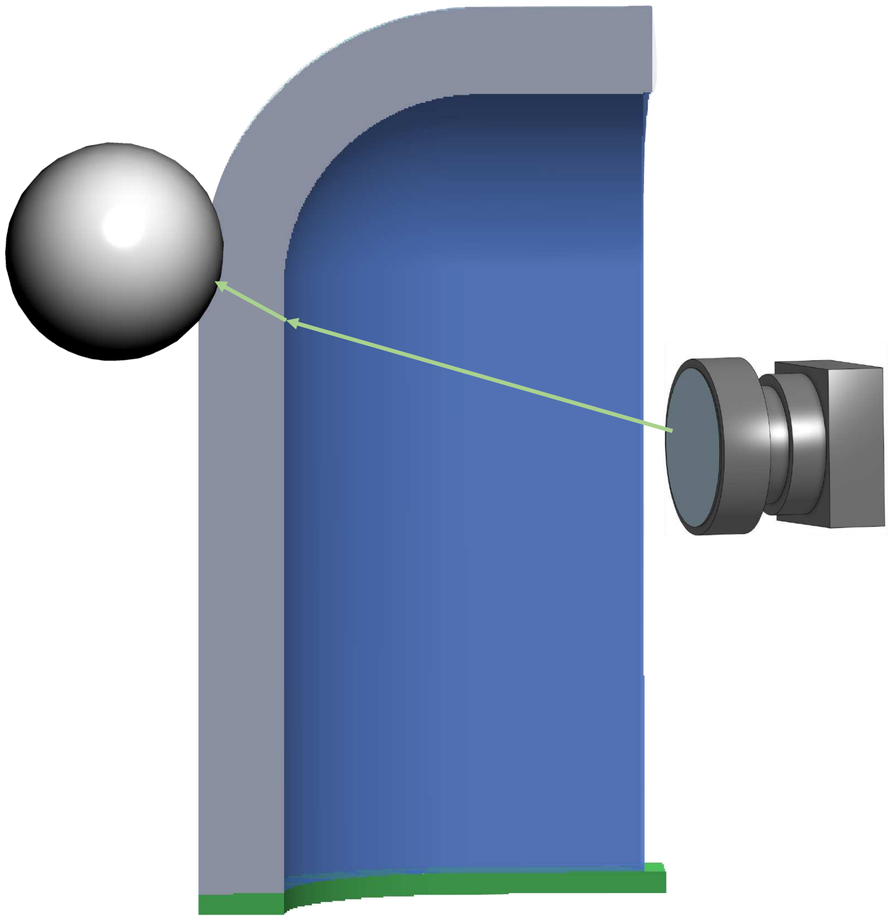
Section View



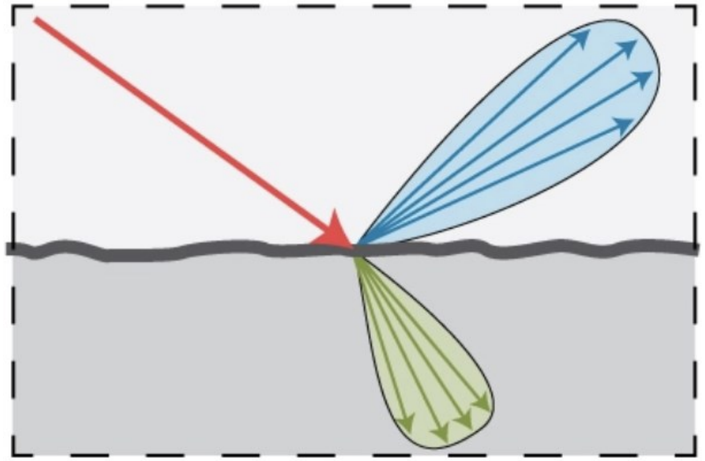
Refraction at rough interface

[Agarwal et al., 2023]

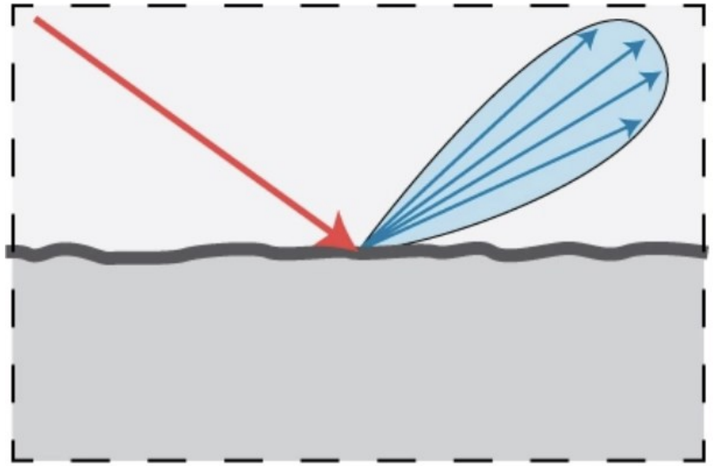
Curved tactile sensor: *working principle*



Section View



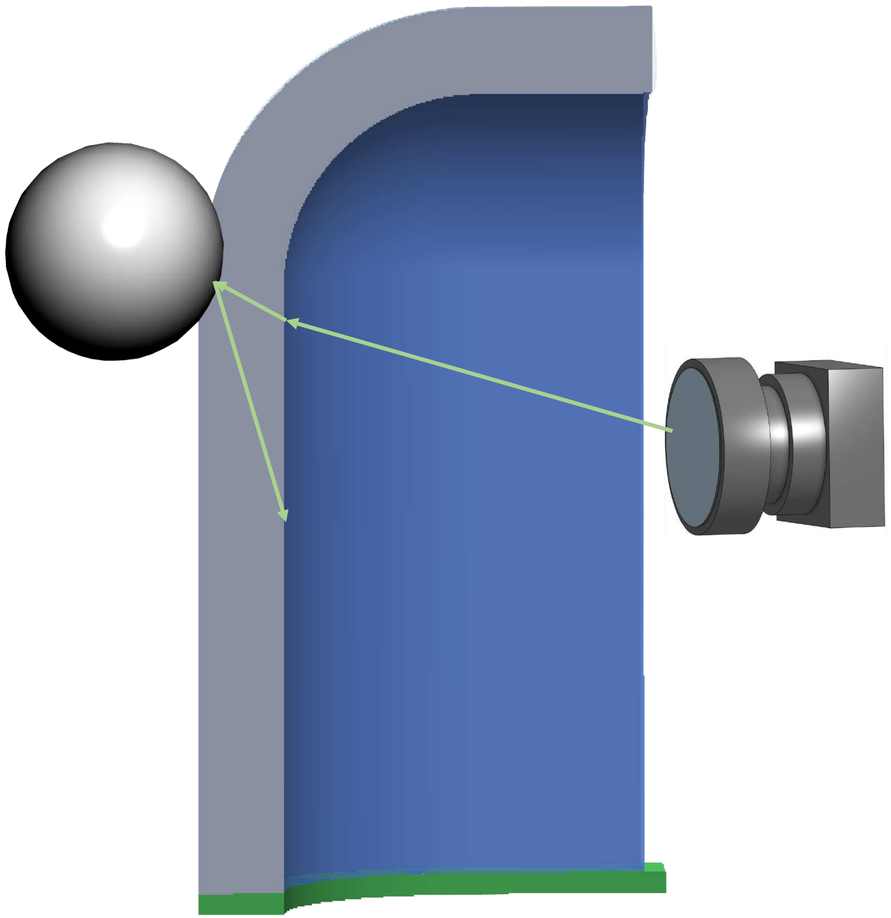
Refraction at rough interface



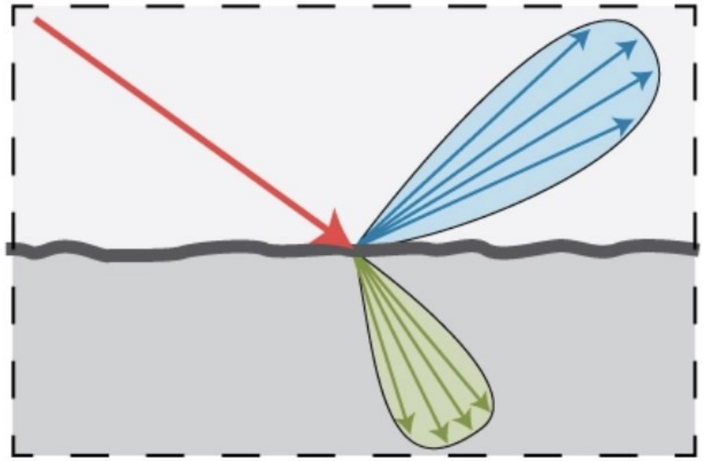
Reflection at glossy surfaces

[Agarwal et al., 2023]

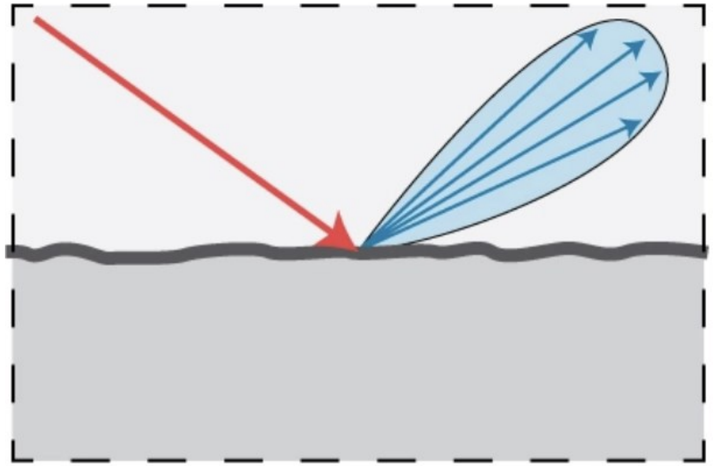
Curved tactile sensor: *working principle*



Section View



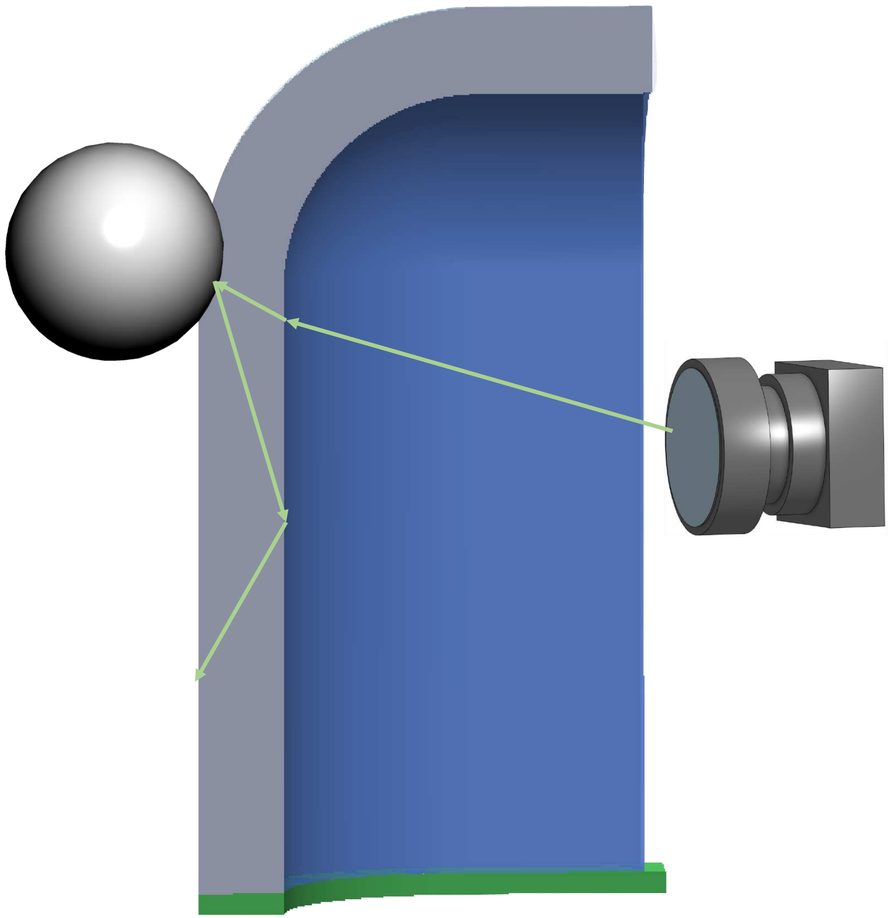
Refraction at rough interface



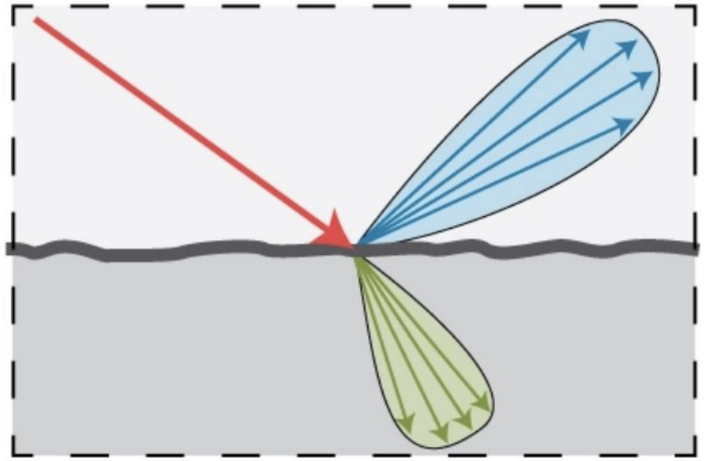
Reflection at glossy surfaces

[Agarwal et al., 2023]

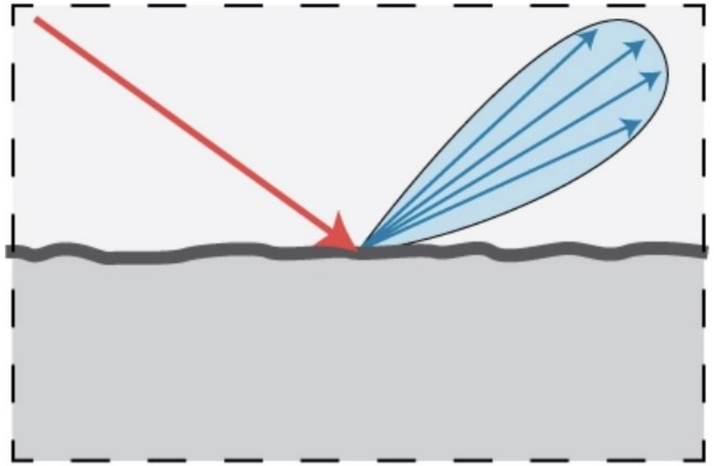
Curved tactile sensor: *working principle*



Section View



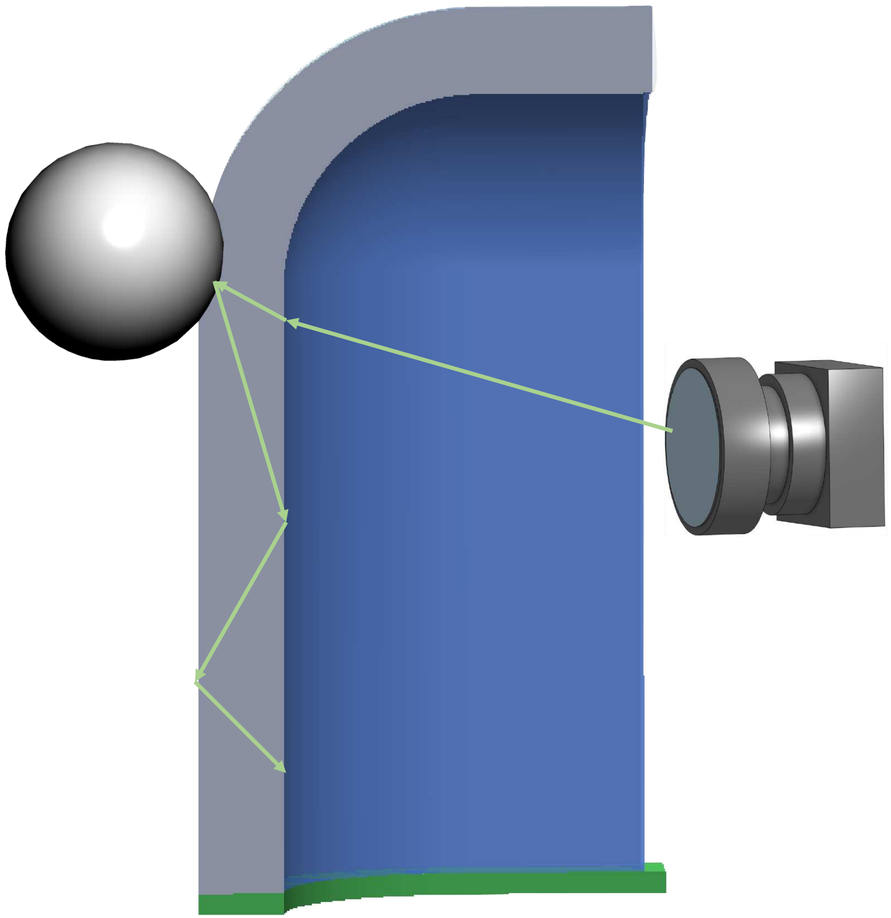
Refraction at rough interface



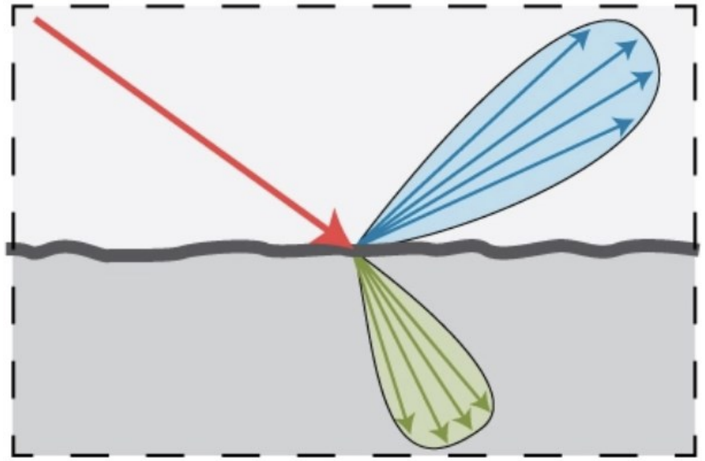
Reflection at glossy surfaces

[Agarwal et al., 2023]

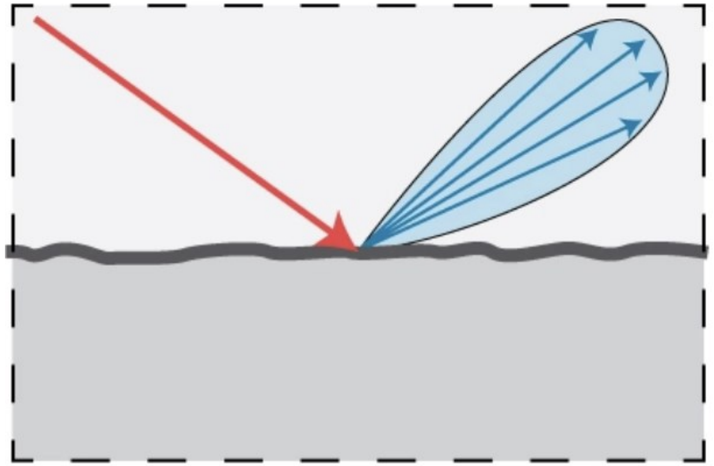
Curved tactile sensor: *working principle*



Section View



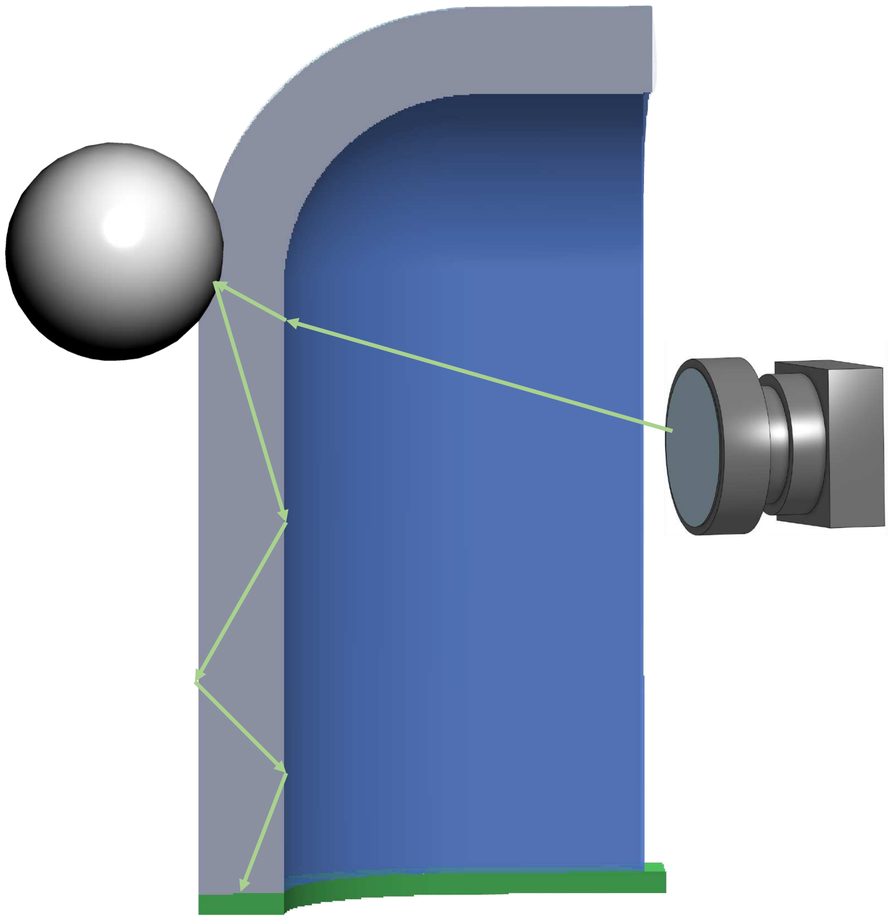
Refraction at rough interface



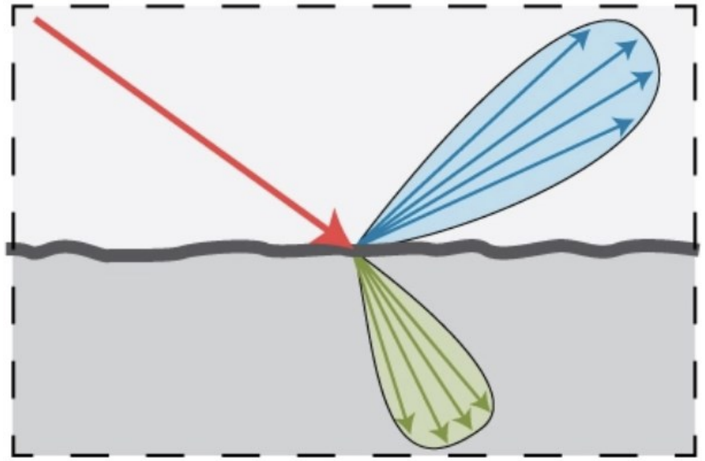
Reflection at glossy surfaces

[Agarwal et al., 2023]

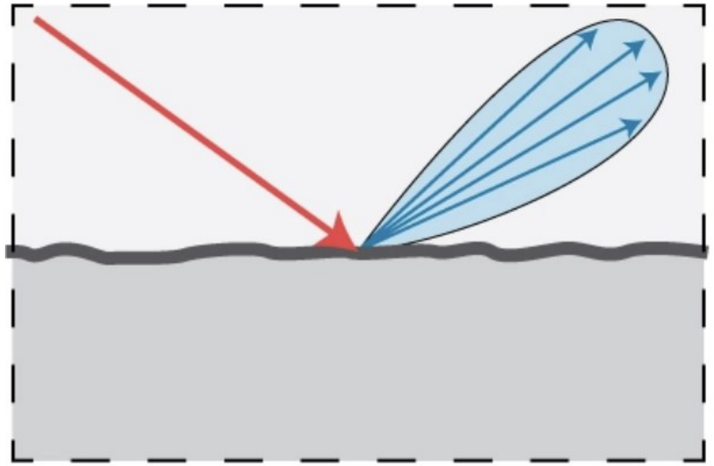
Curved tactile sensor: *working principle*



Section View



Refraction at rough interface

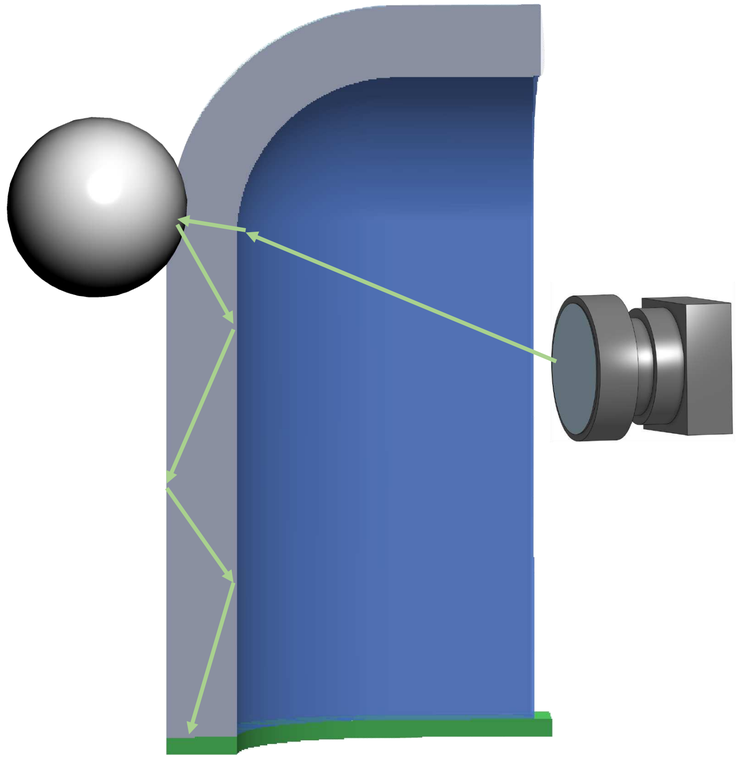


Reflection at glossy surfaces

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- Sensor geometry
- Material specifications
- Placement of camera and lights

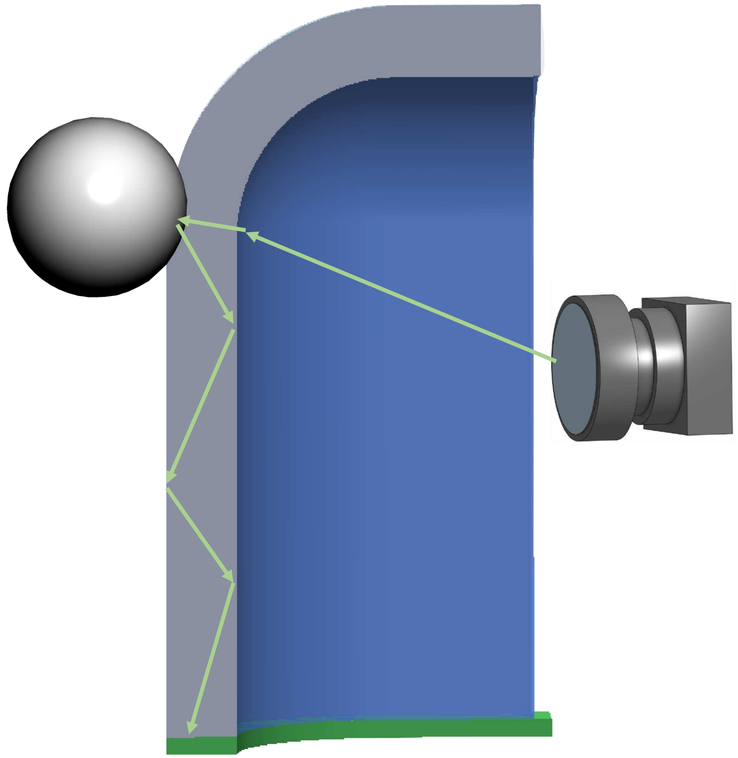


Section view

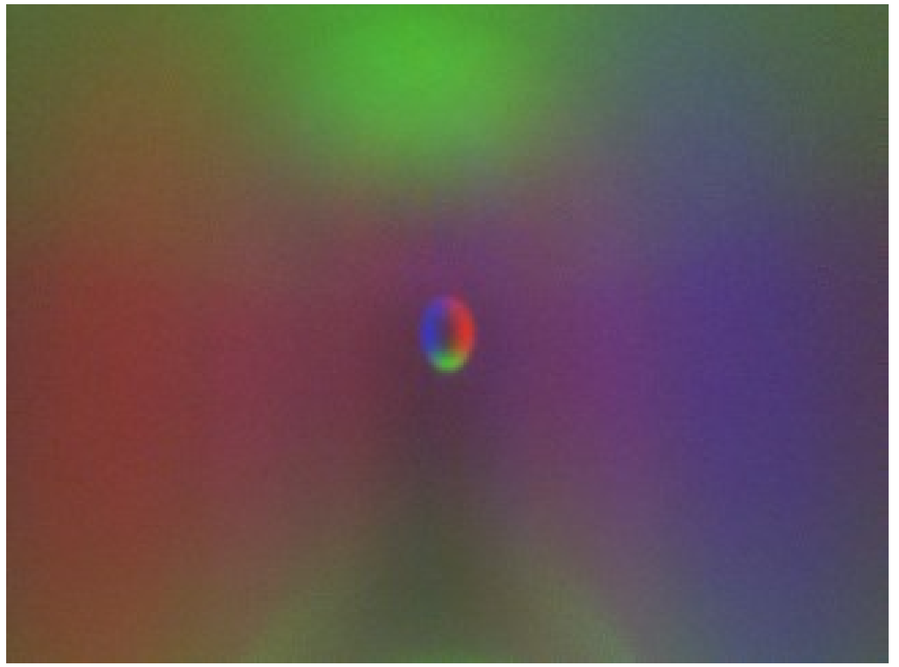
Tactile image [Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- Sensor geometry
- Material specifications
- Placement of camera and lights



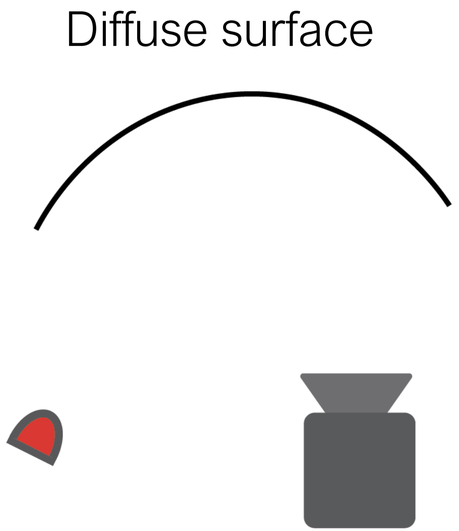
Section view



Tactile image [Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



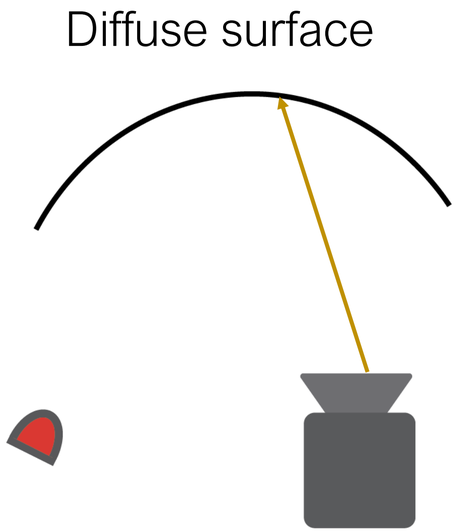
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



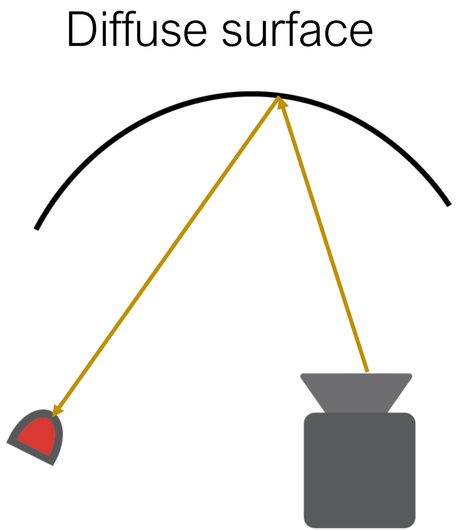
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



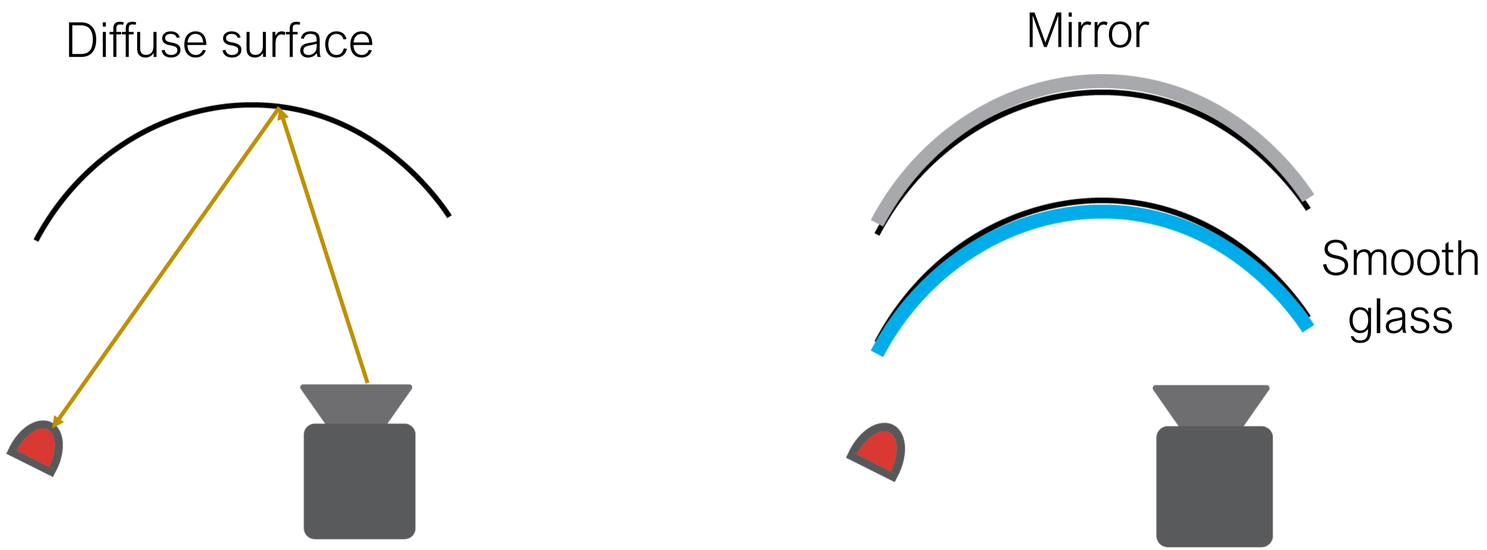
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



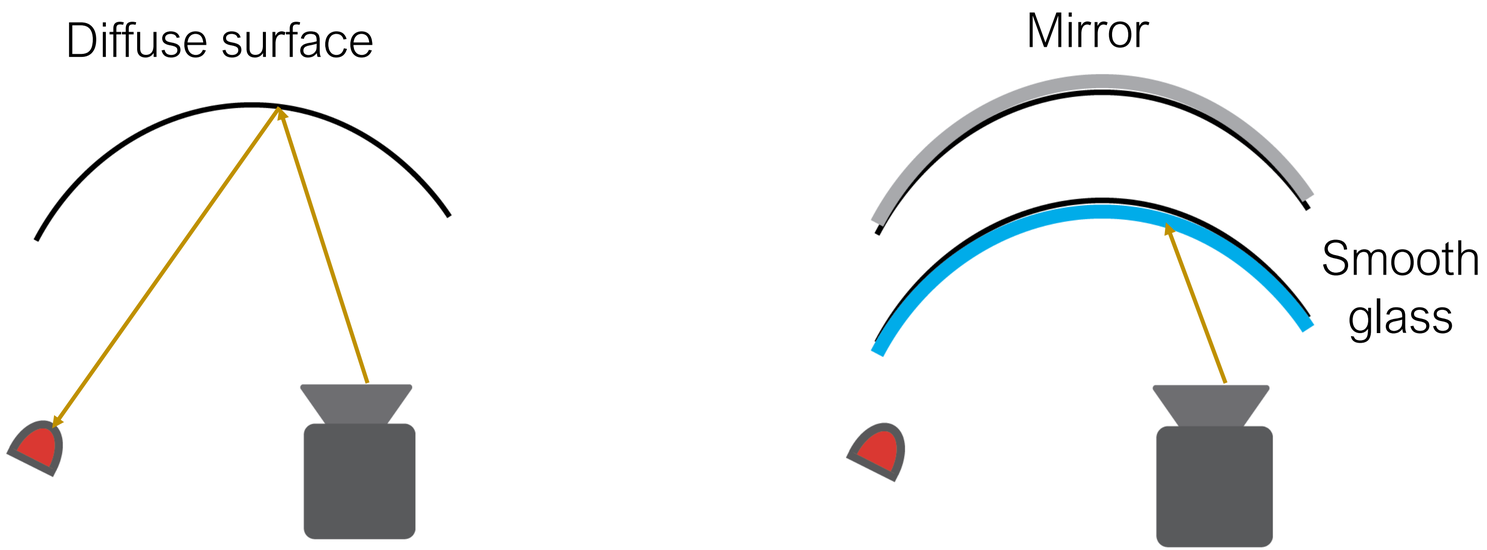
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



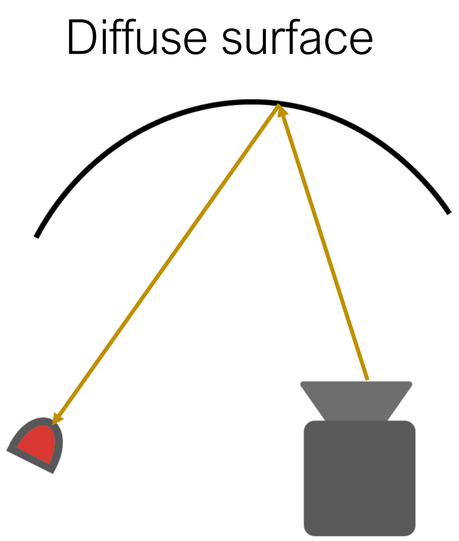
(Relatively) easy cases

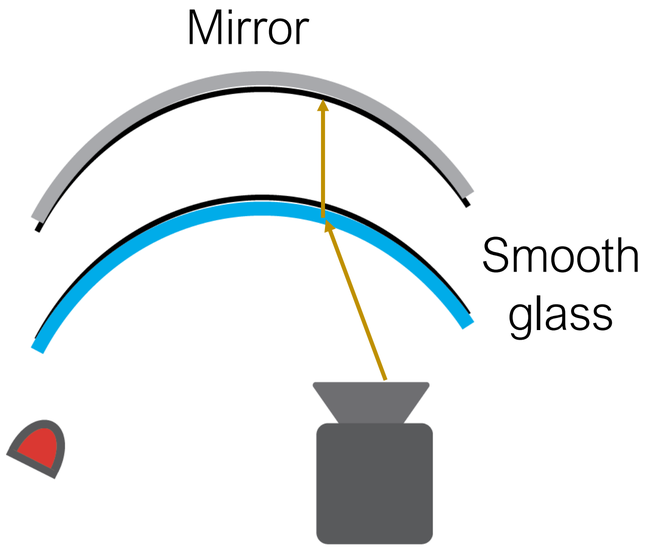
Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging





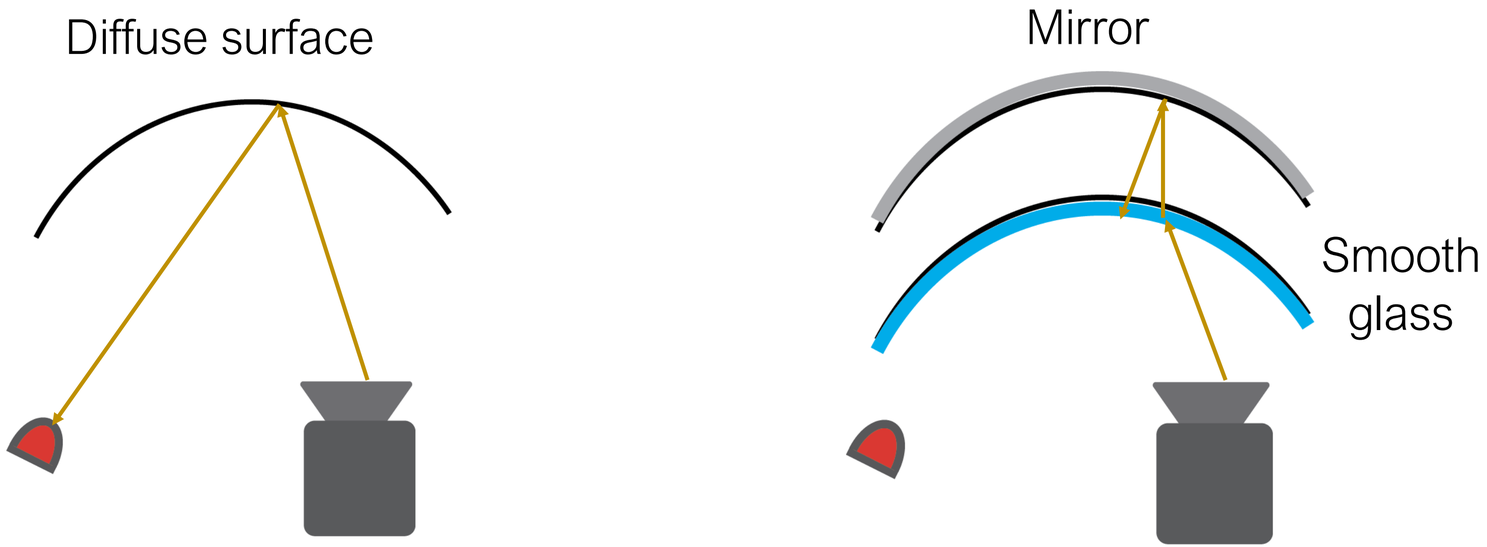
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



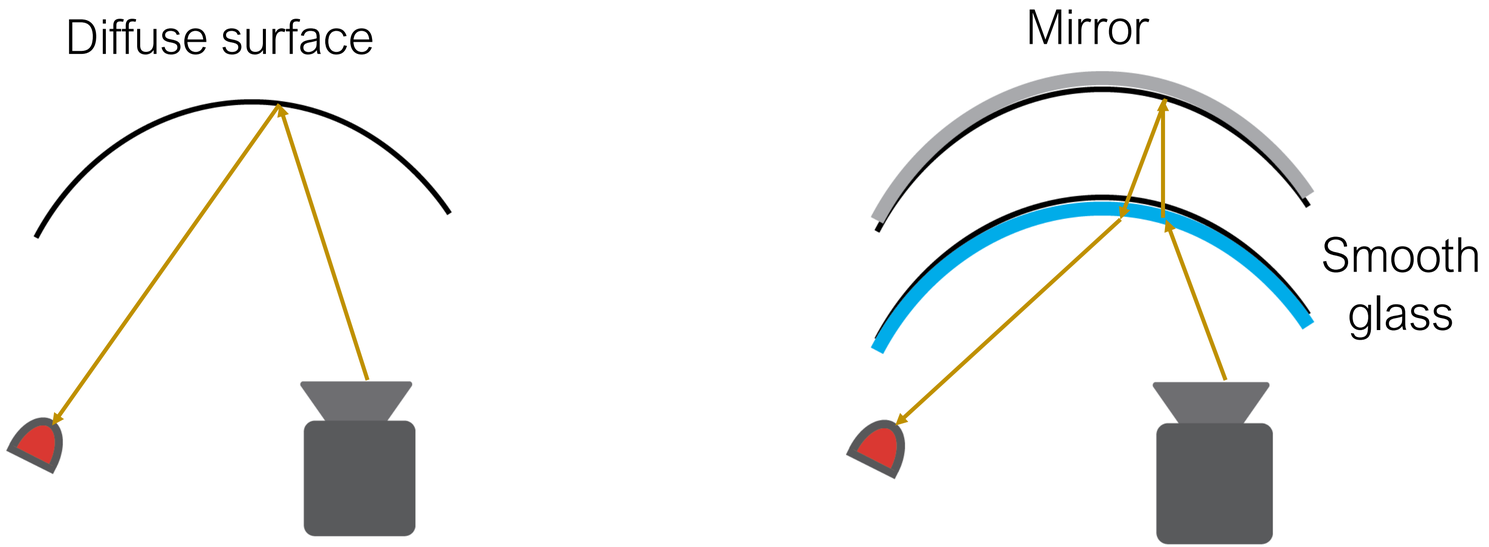
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging



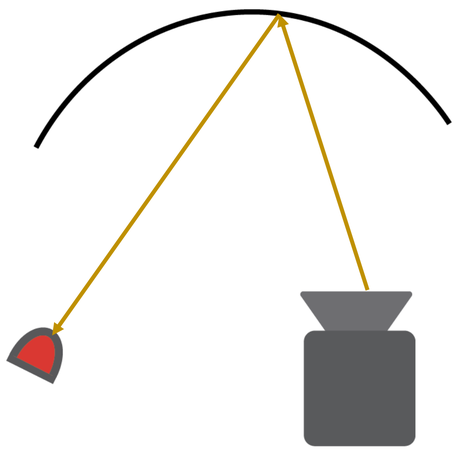
(Relatively) easy cases

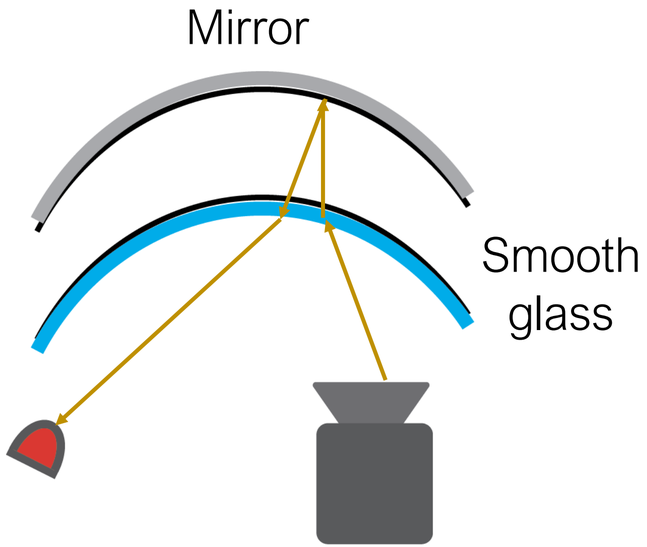
Hard case

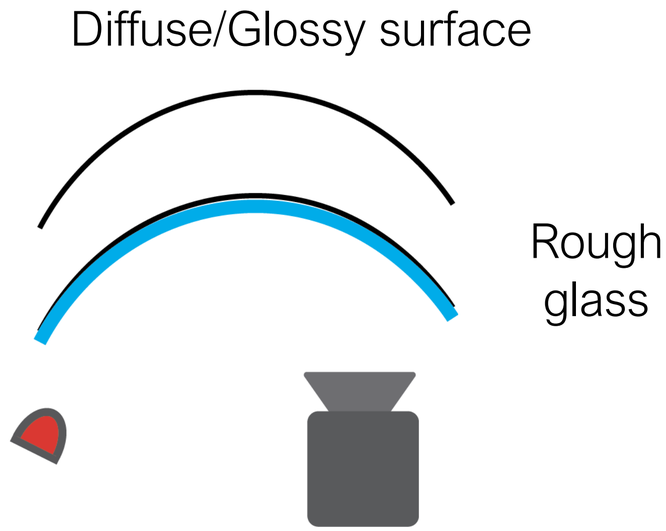
[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging







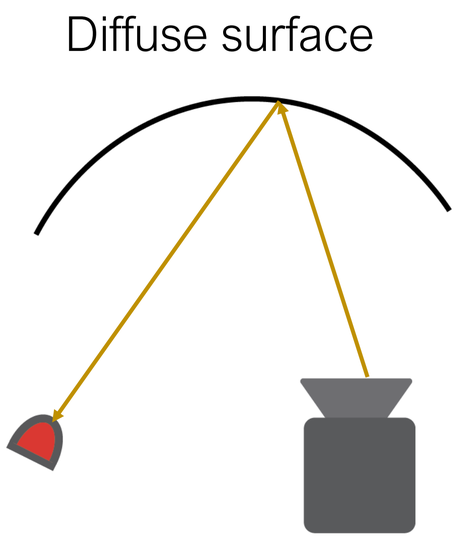
(Relatively) easy cases

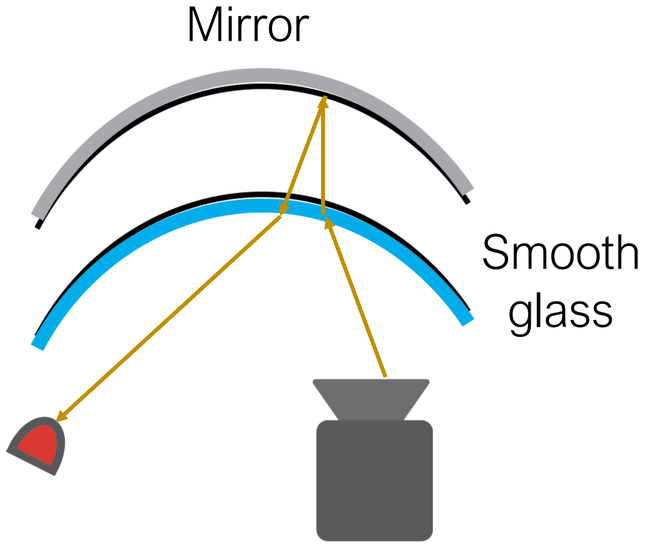
Hard case

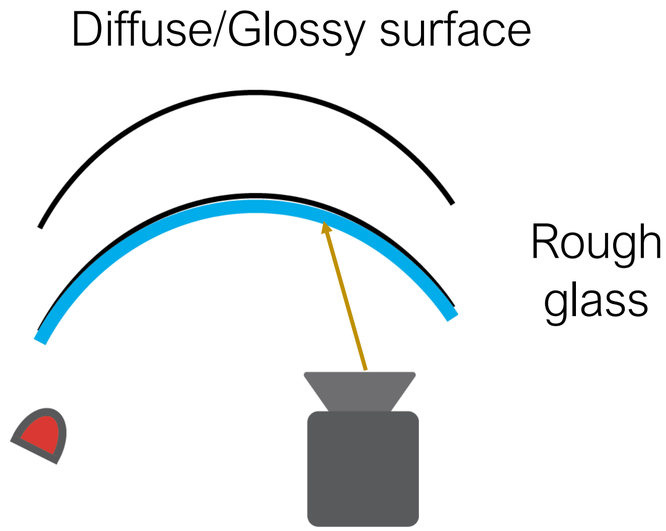
[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging







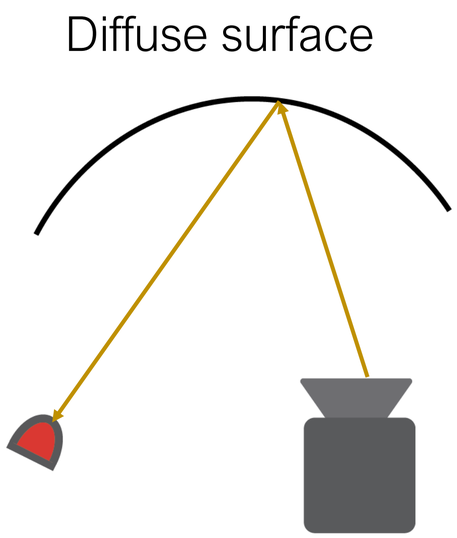
(Relatively) easy cases

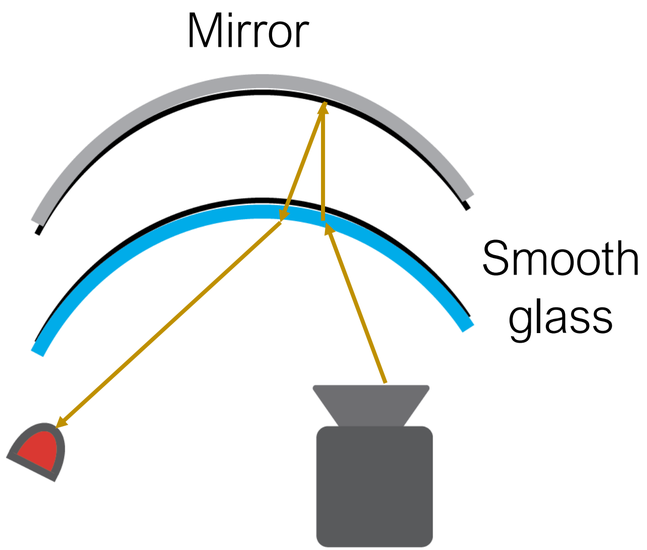
Hard case

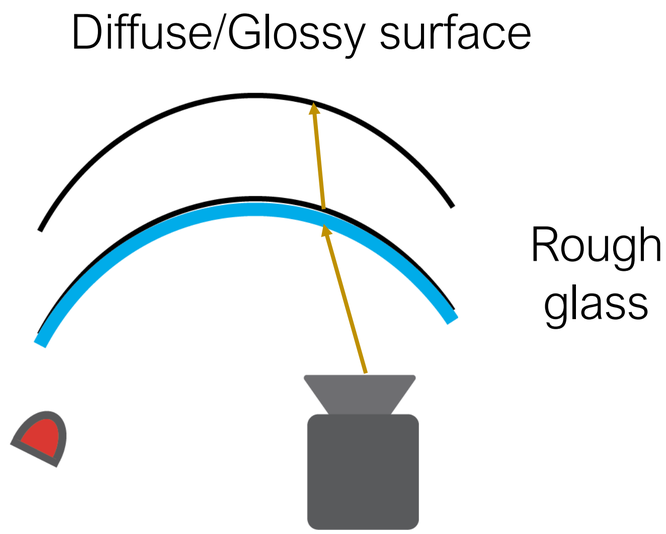
[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging







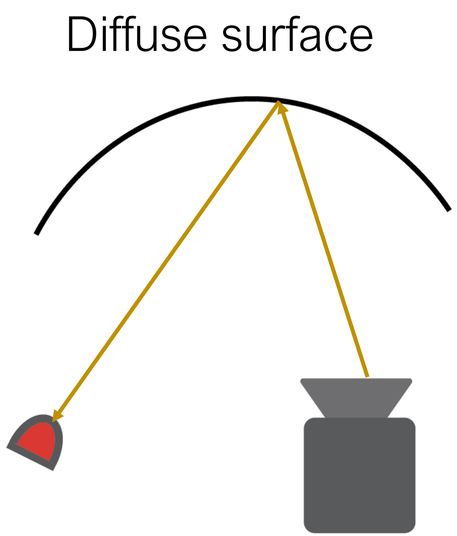
(Relatively) easy cases

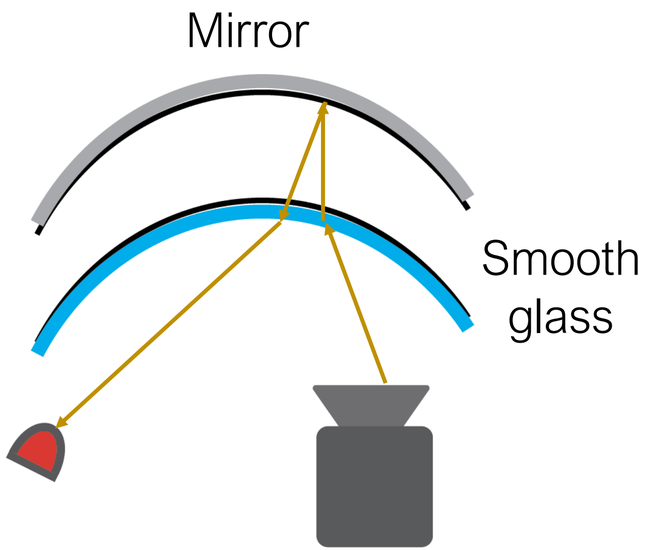
Hard case

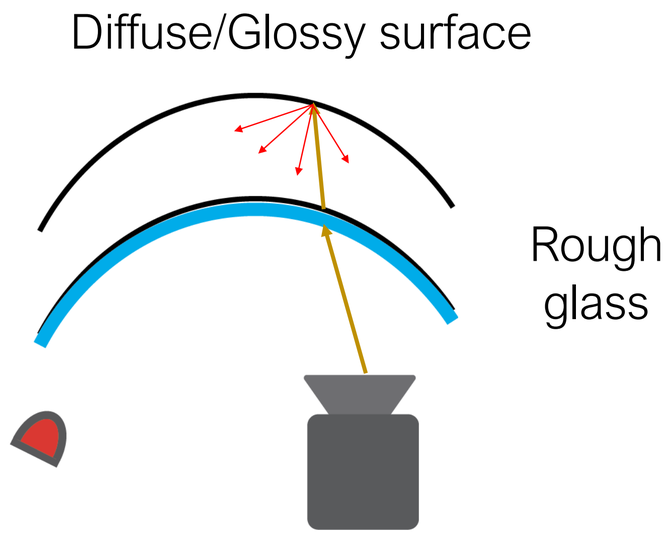
[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging







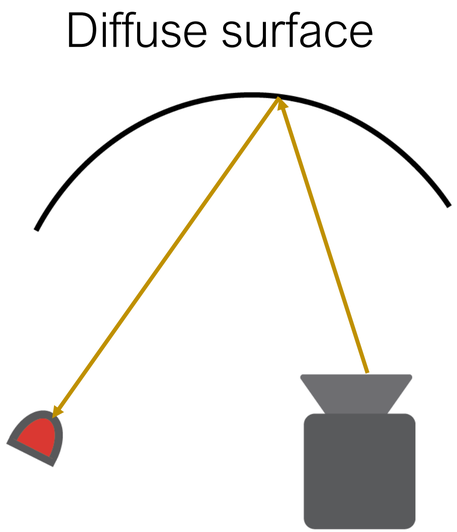
(Relatively) easy cases

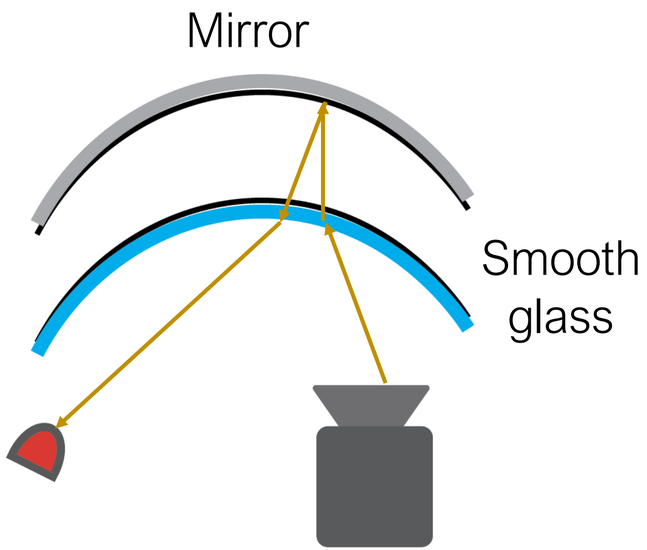
Hard case

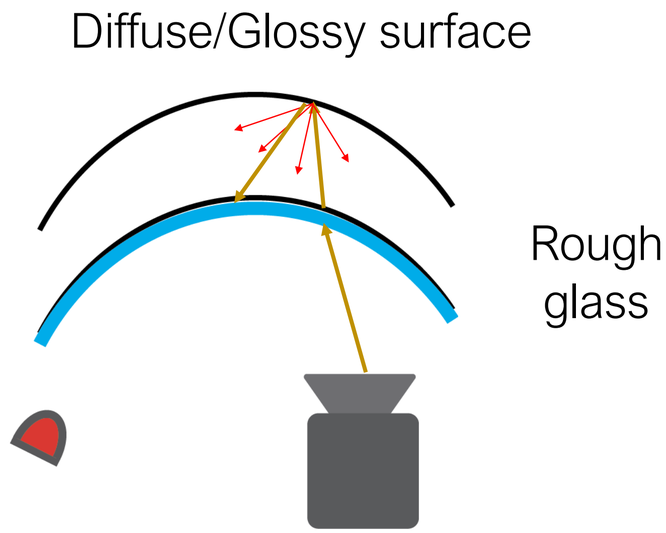
[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging







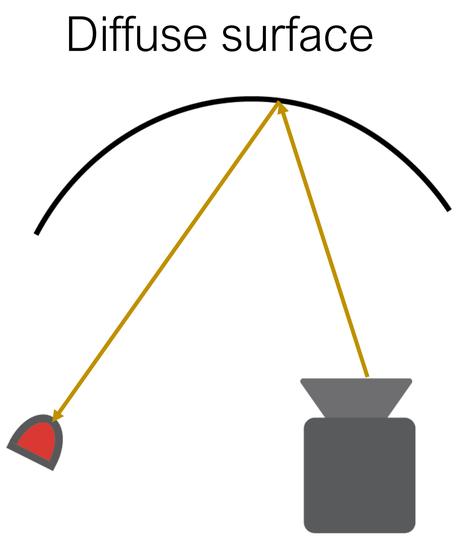
(Relatively) easy cases

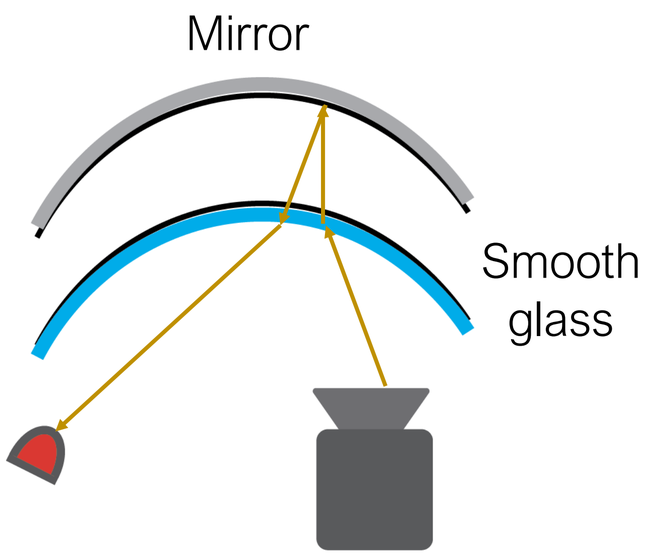
Hard case

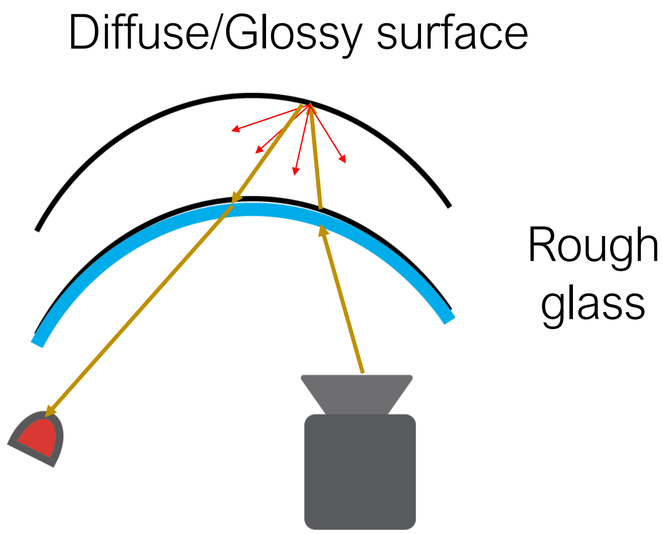
[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging







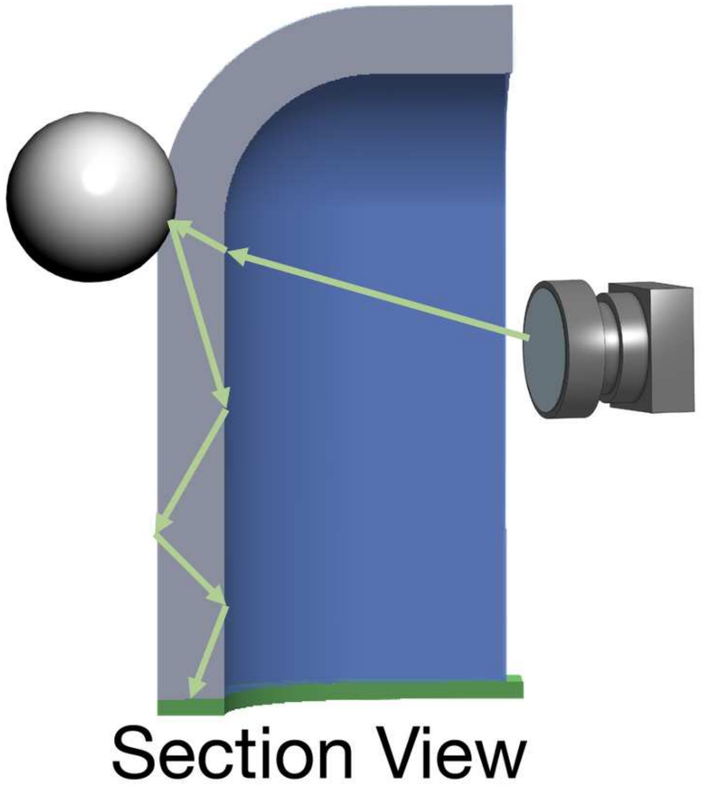
(Relatively) easy cases

Hard case

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

- What makes simulation challenging
  - SDS light paths
  - Indirect illumination



[Agarwal et al., 2023]

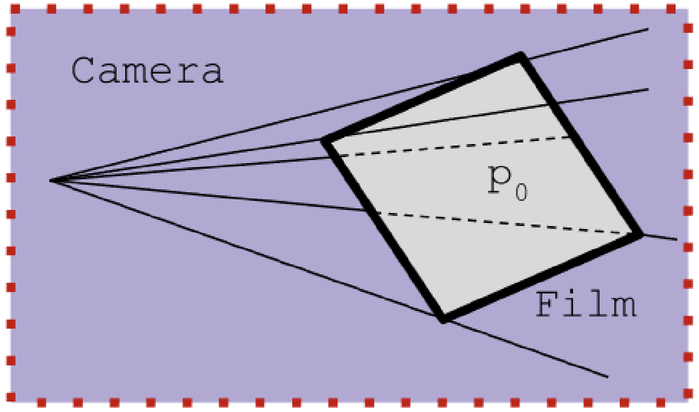
Sensor design framework: *optical simulation*

- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea → Slowly mutate paths to generate useful paths

[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

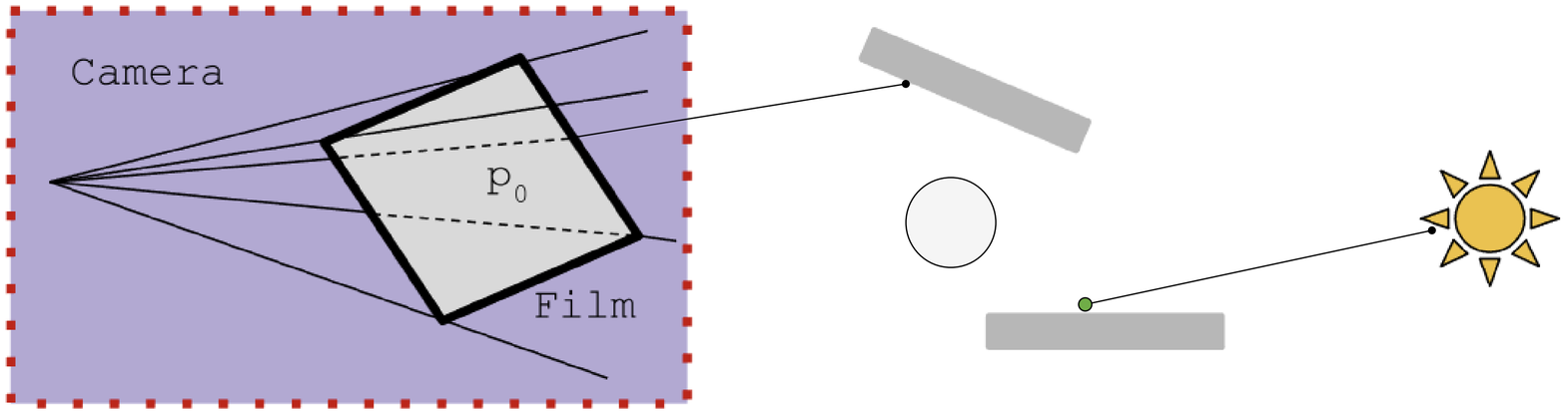
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

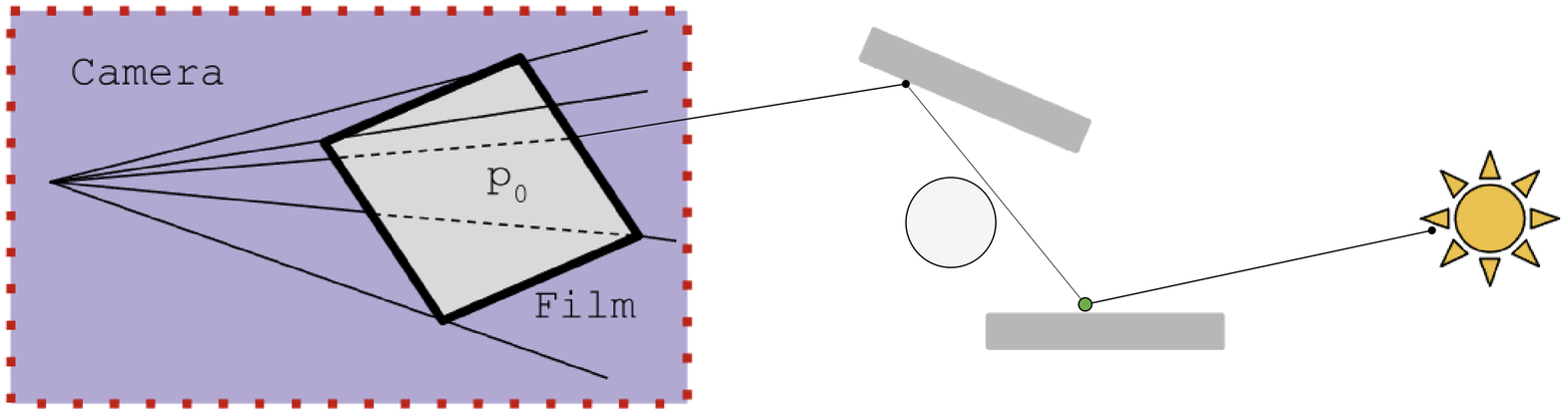
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

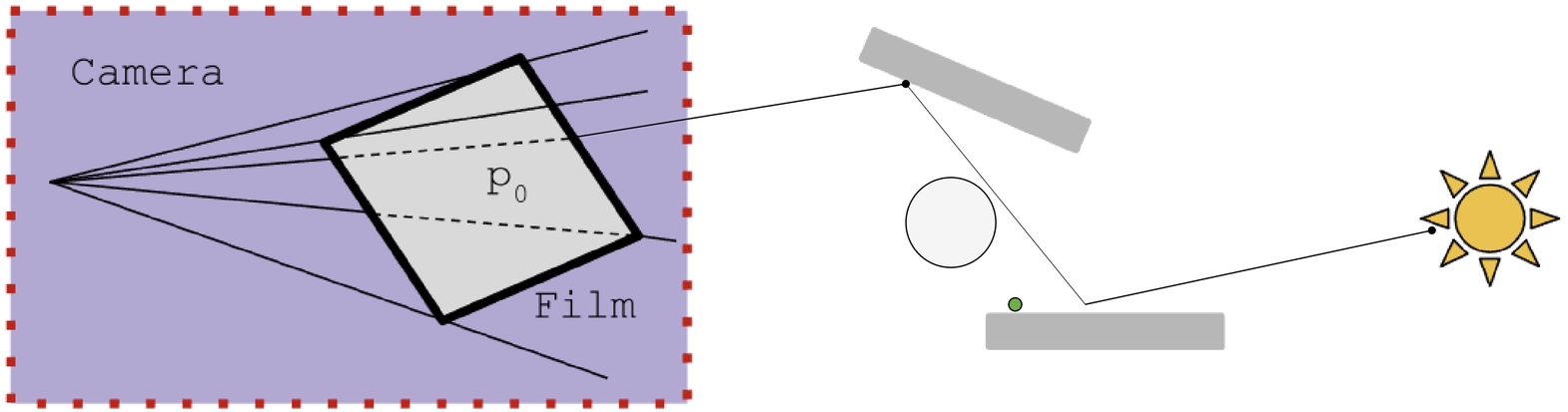
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

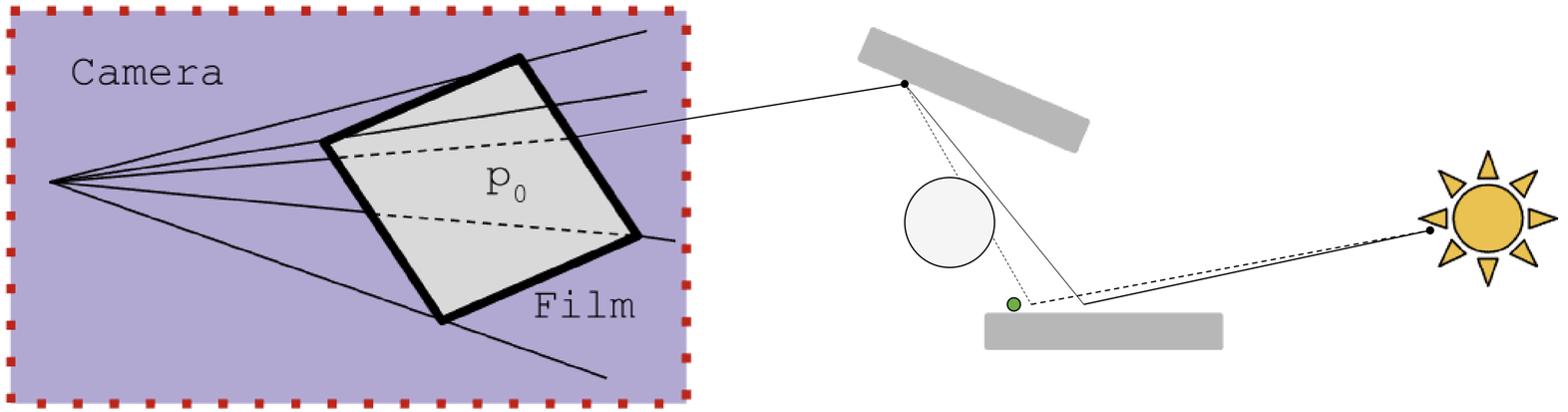
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

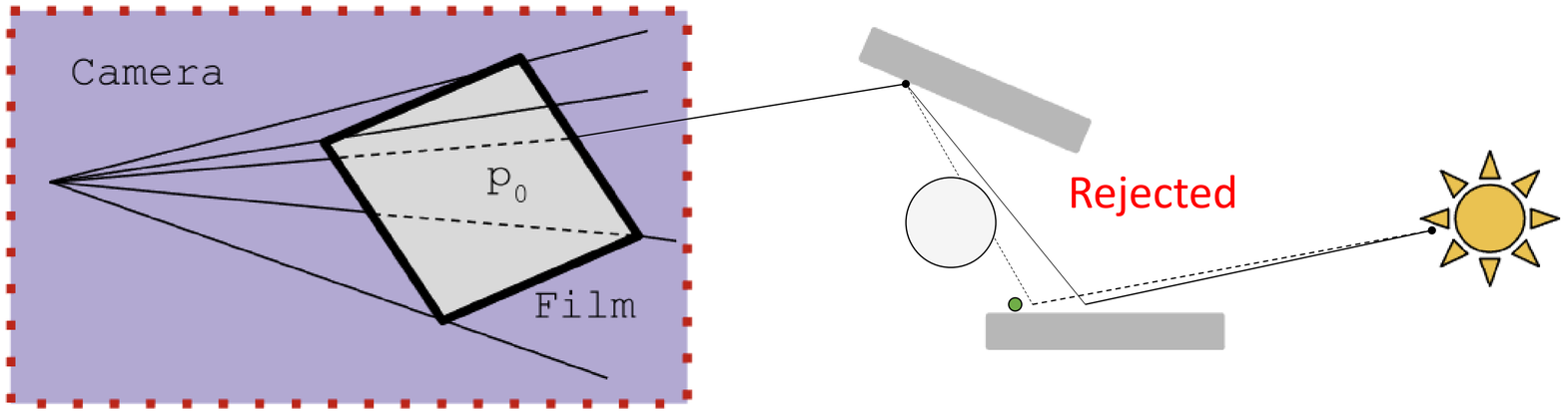
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

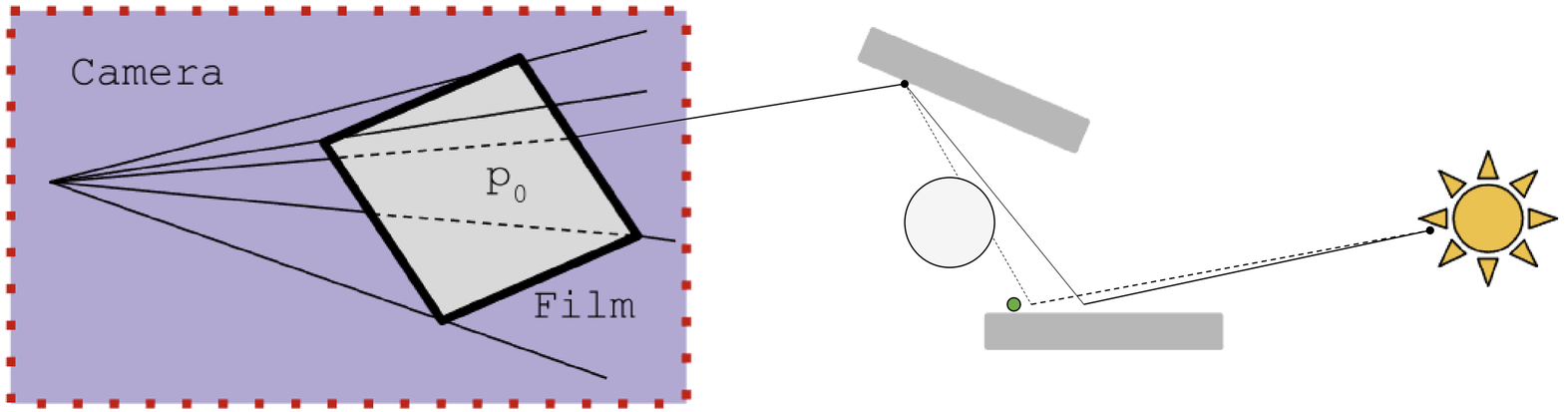
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

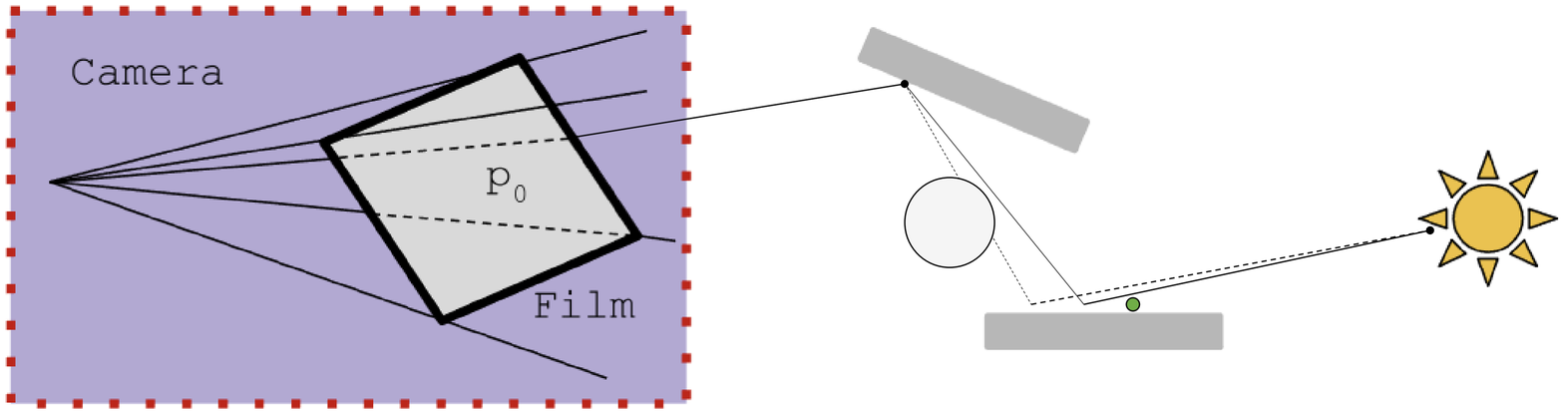
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

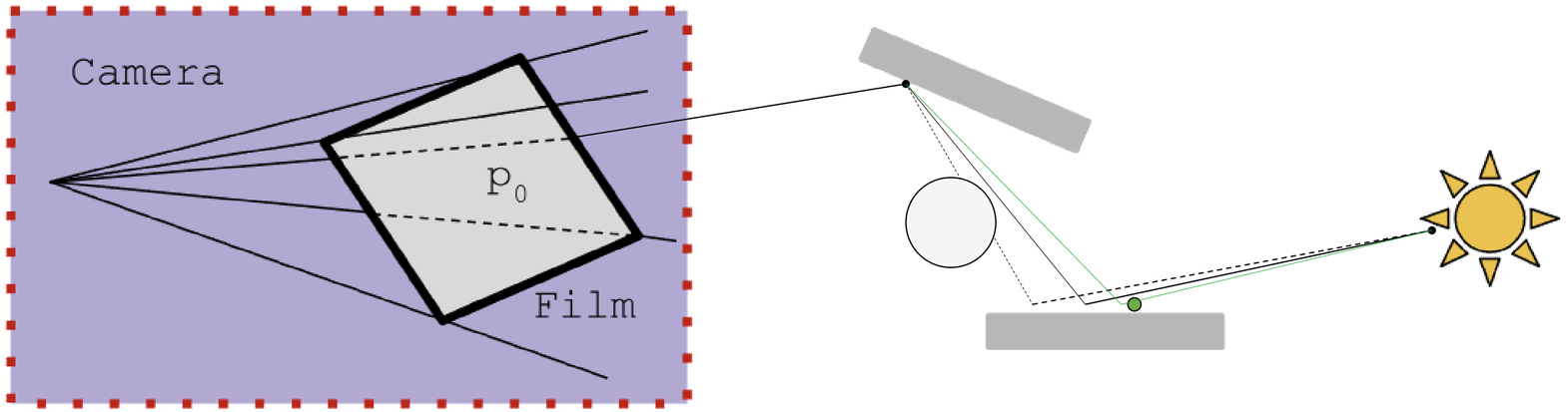
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

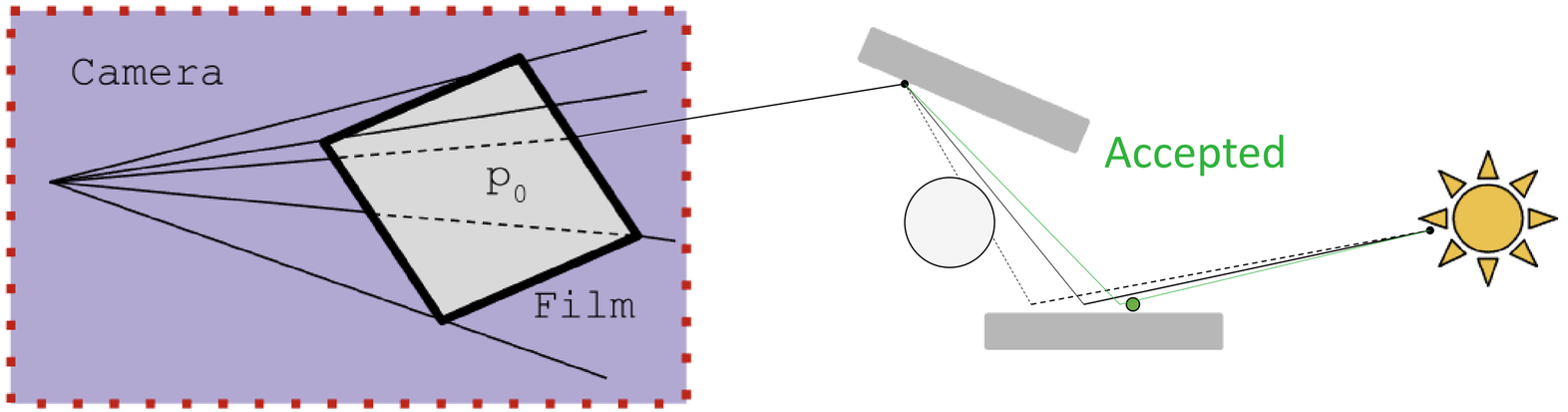
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

Sensor design framework: *optical simulation*

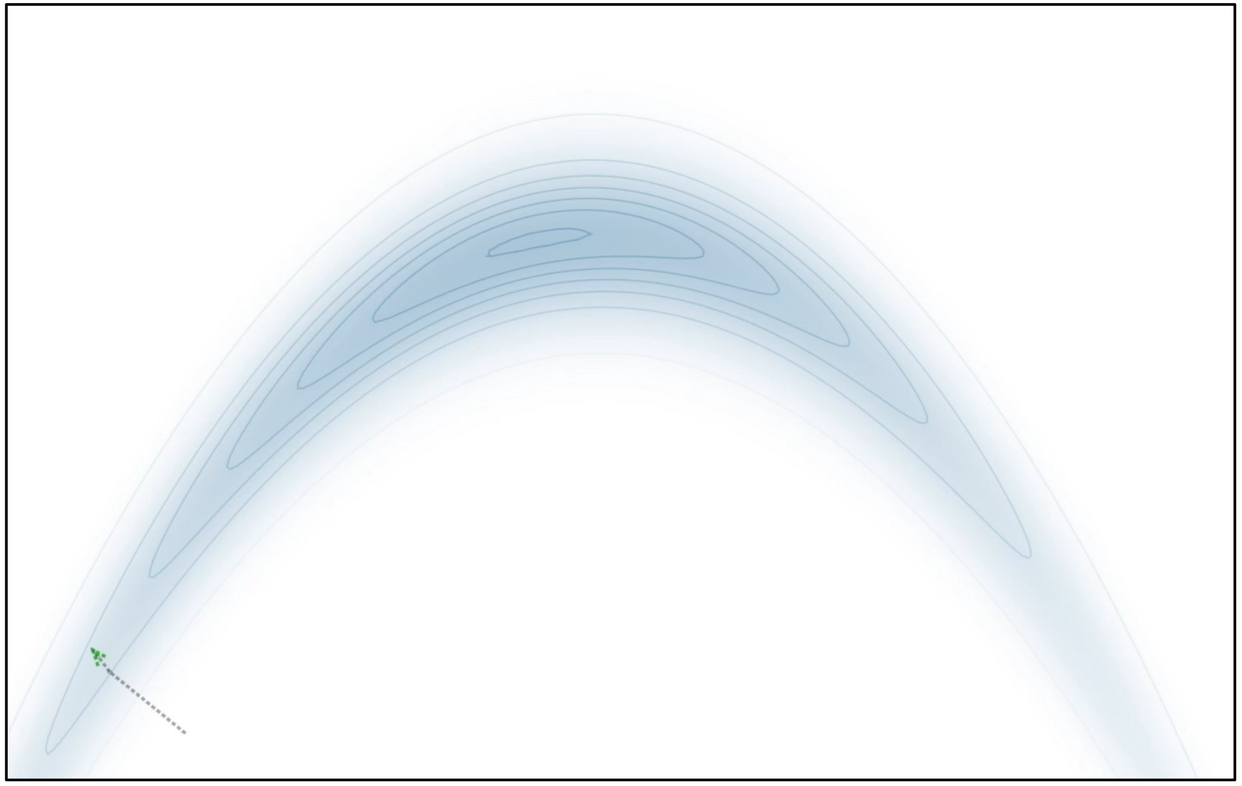
- What makes simulation challenging
  - SDS light paths
  - Indirect illumination
- Require Markov Chain Monte Carlo rendering techniques
  - Key idea ➡ Slowly mutate paths to generate useful paths



[Agarwal et al., 2023]

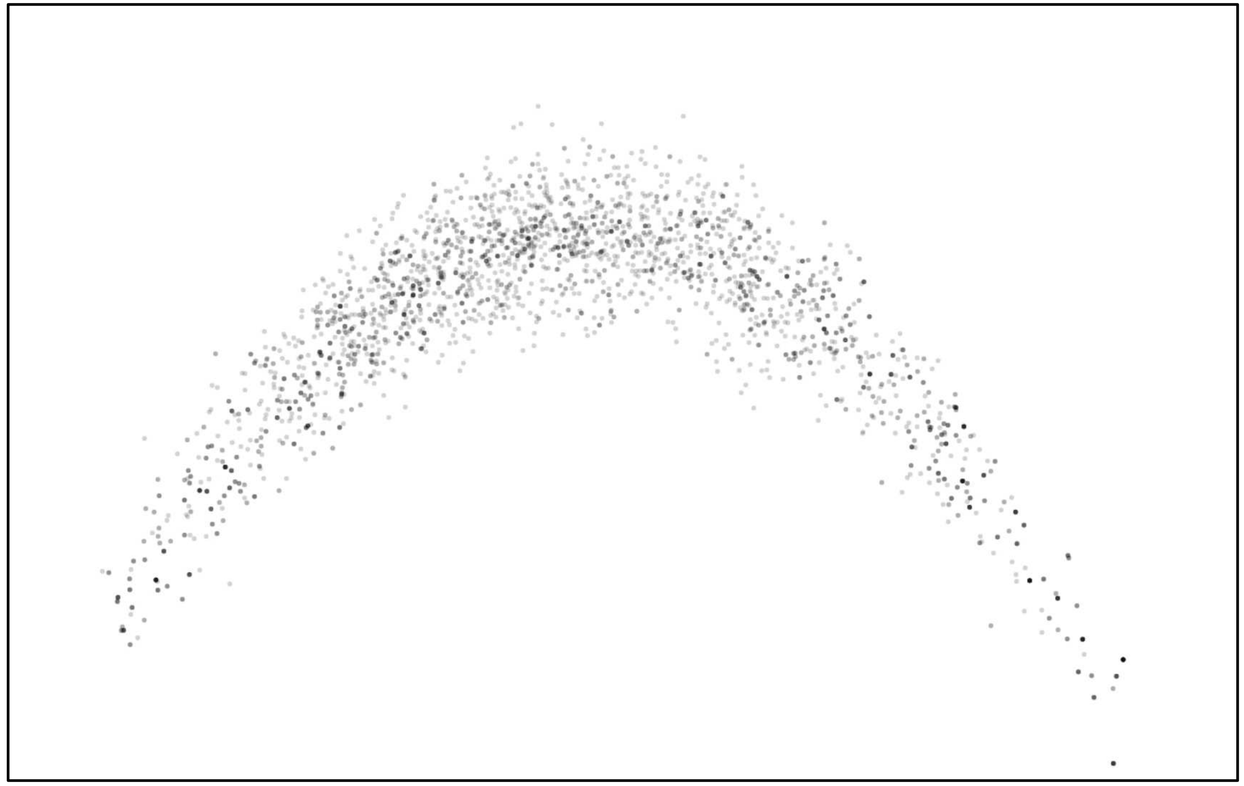
Optimization:

- Stochastic Gradient Descent (SGD)



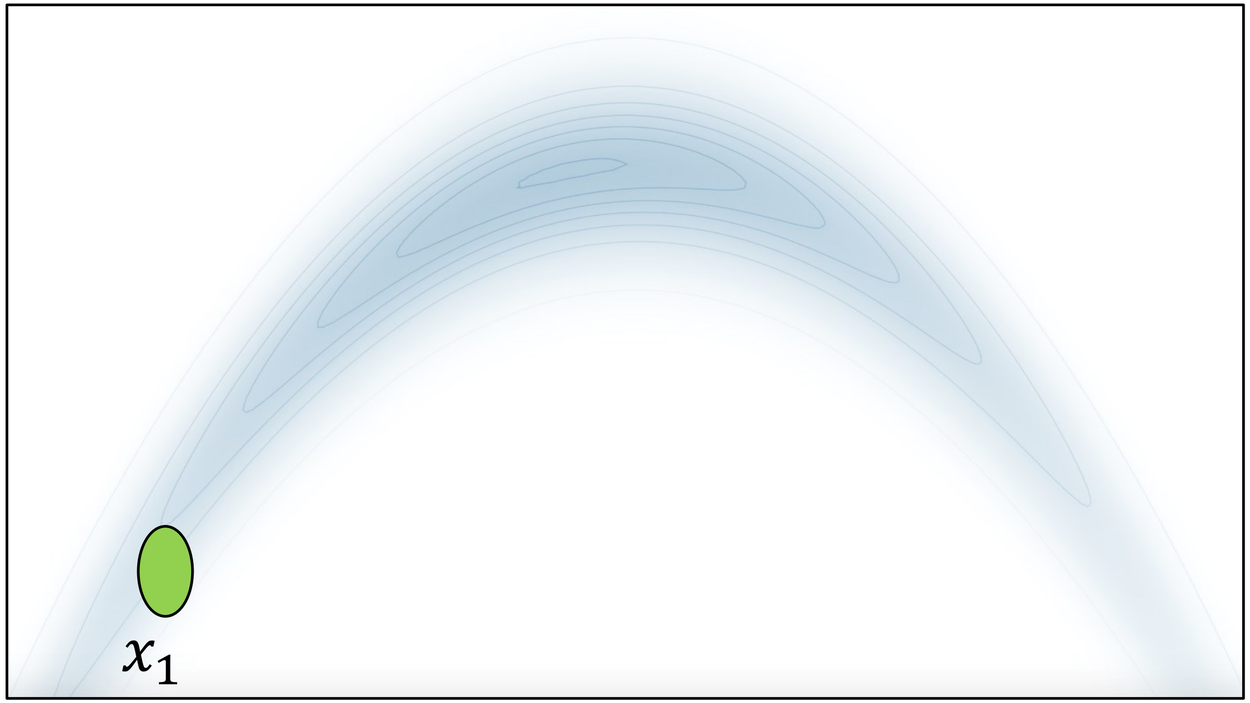
MCMC sampling:

- Langevin Monte Carlo (LMC)



[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

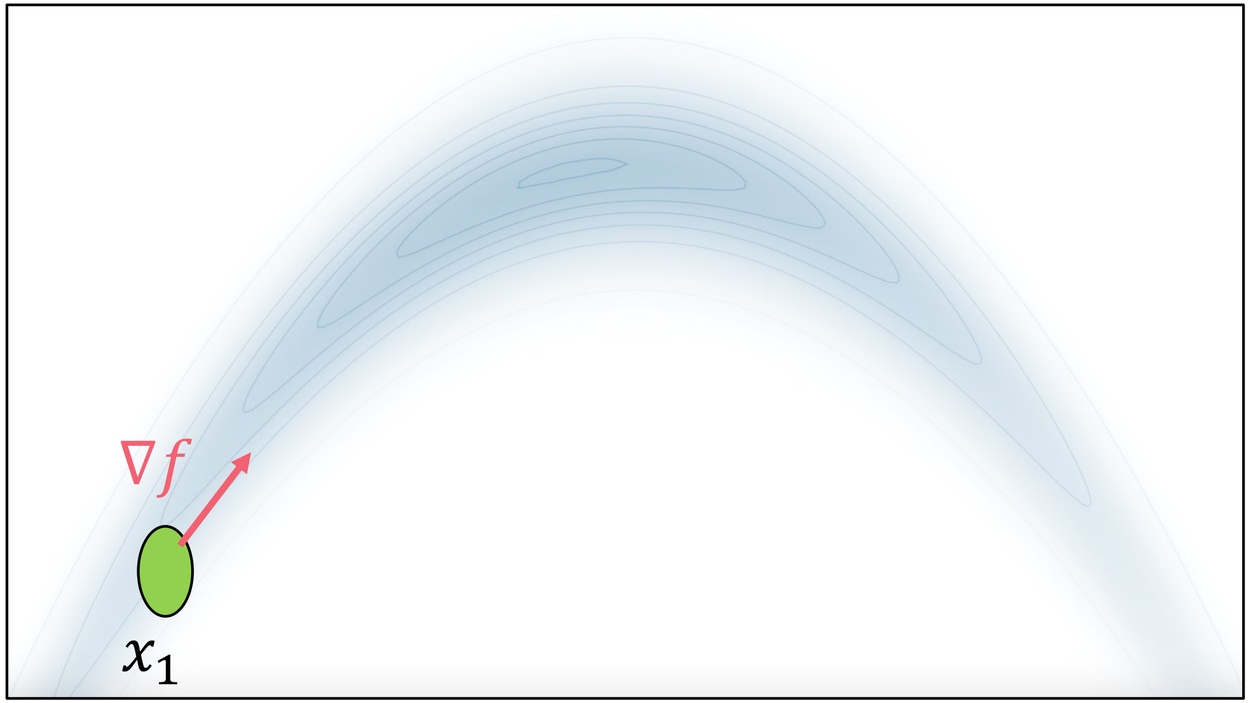
$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

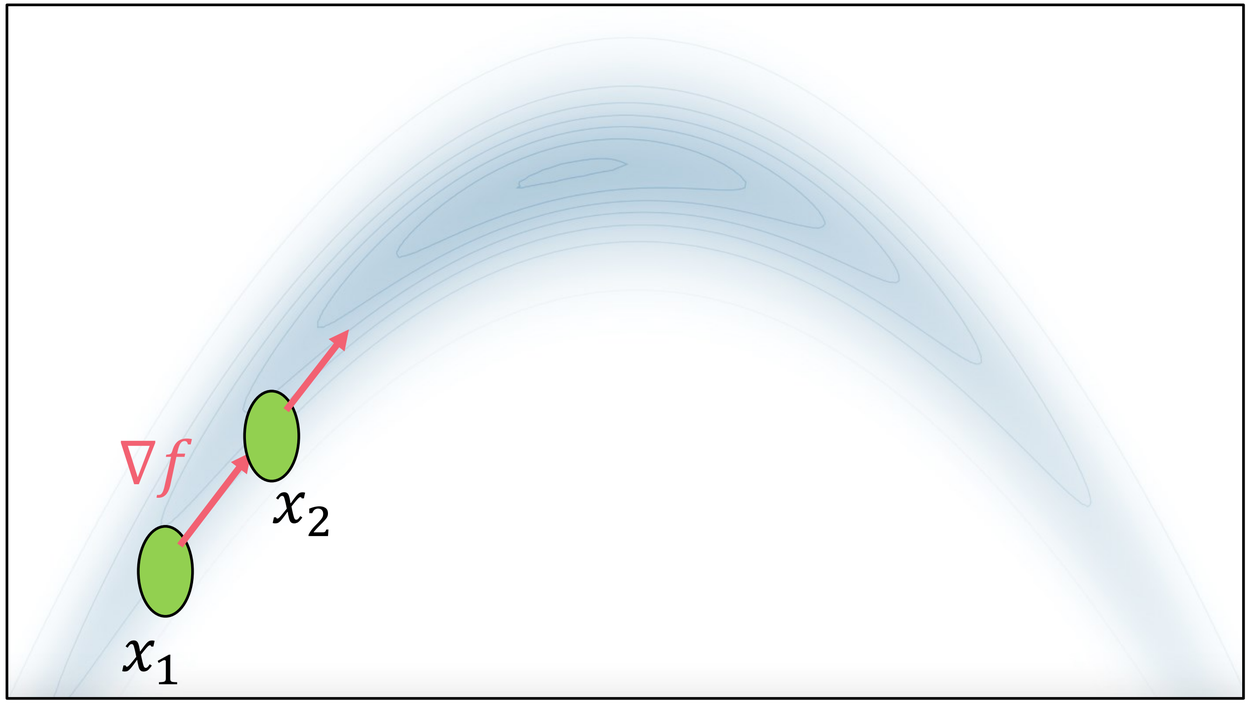
$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

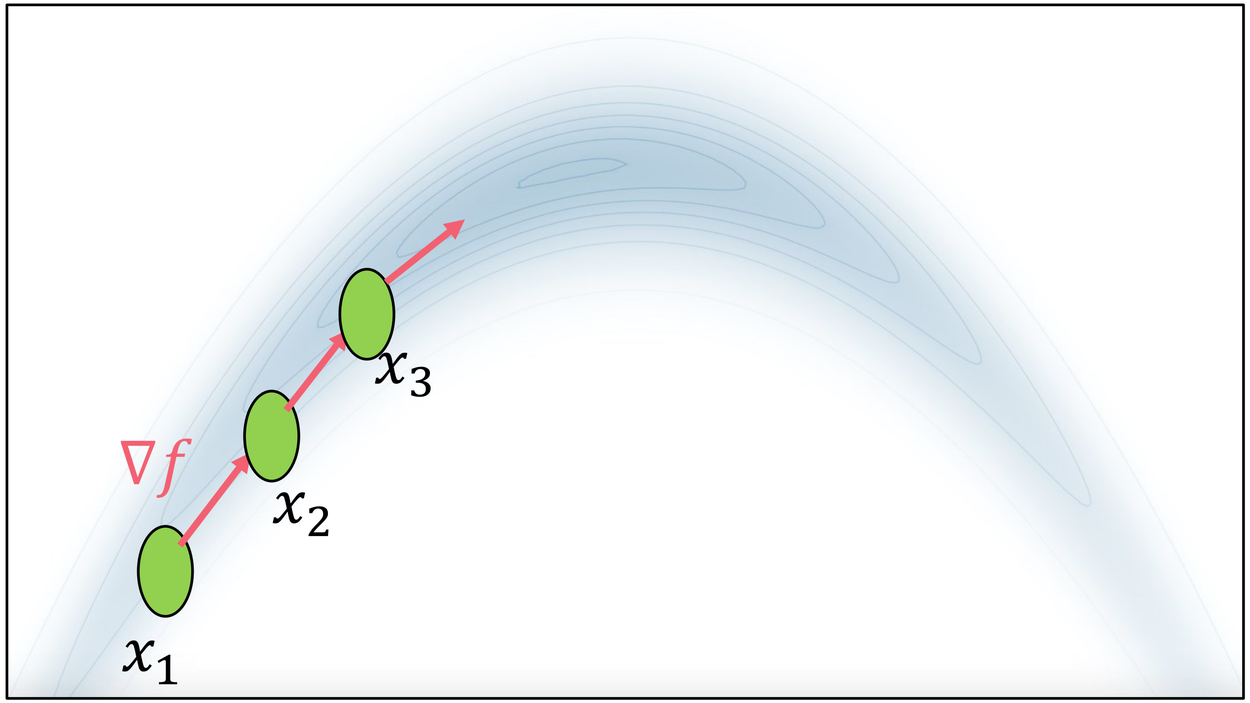
$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

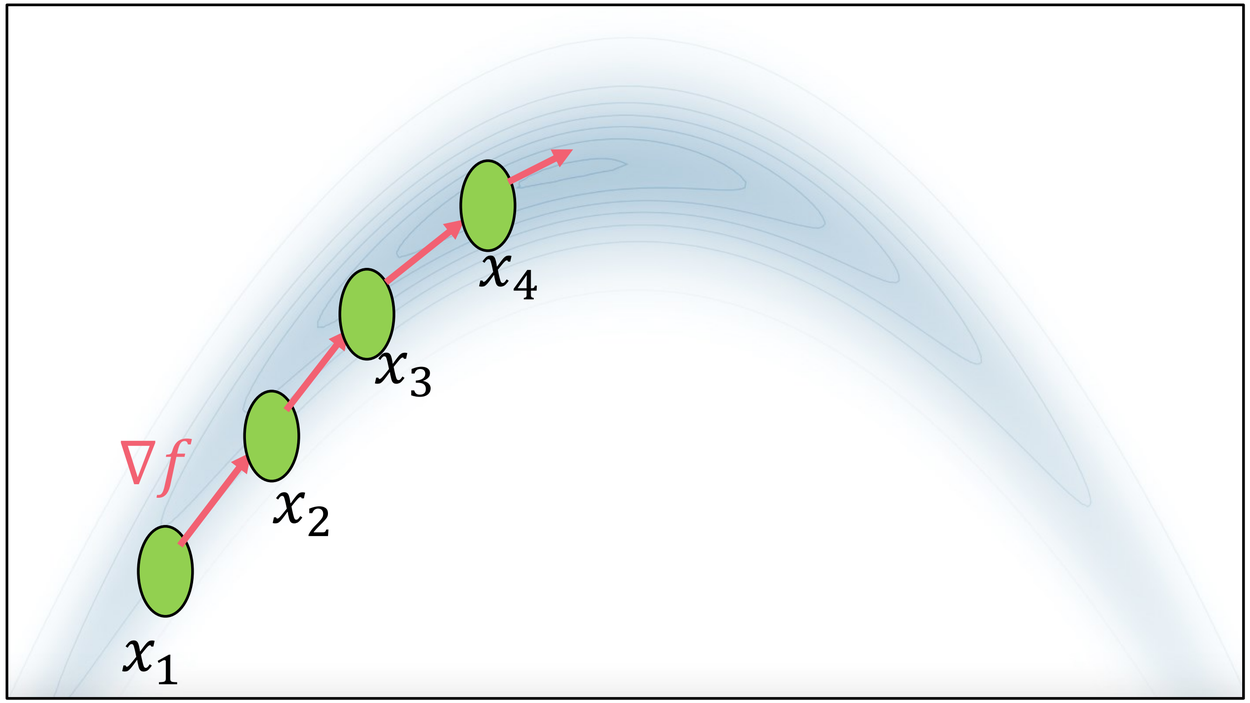
$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

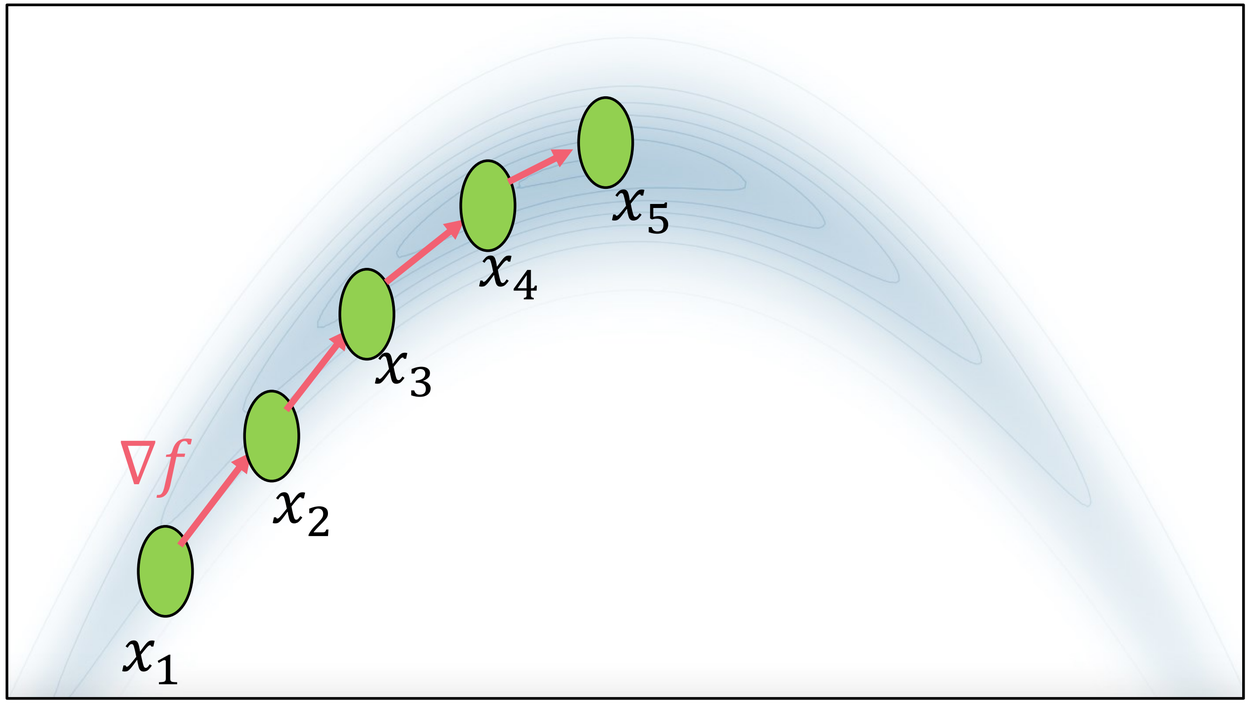
$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

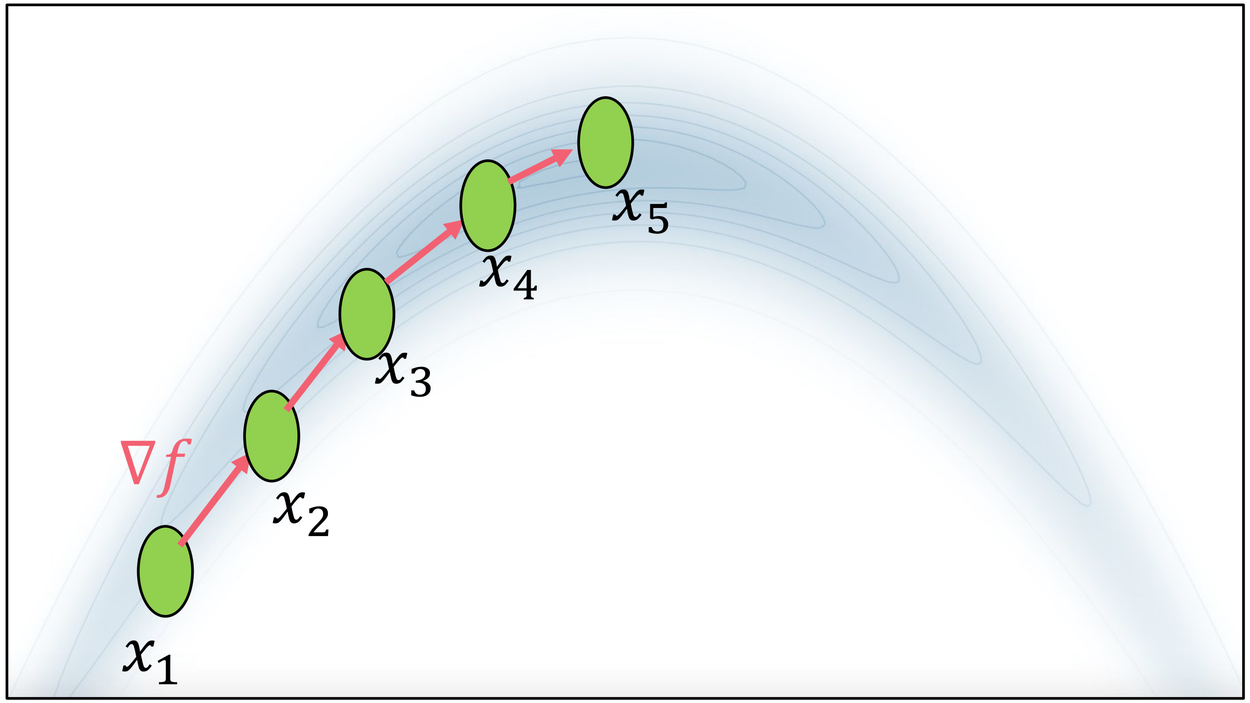
$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

[Luan et al., 2020]

GD OVERVIEW



Optimization problem:

$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

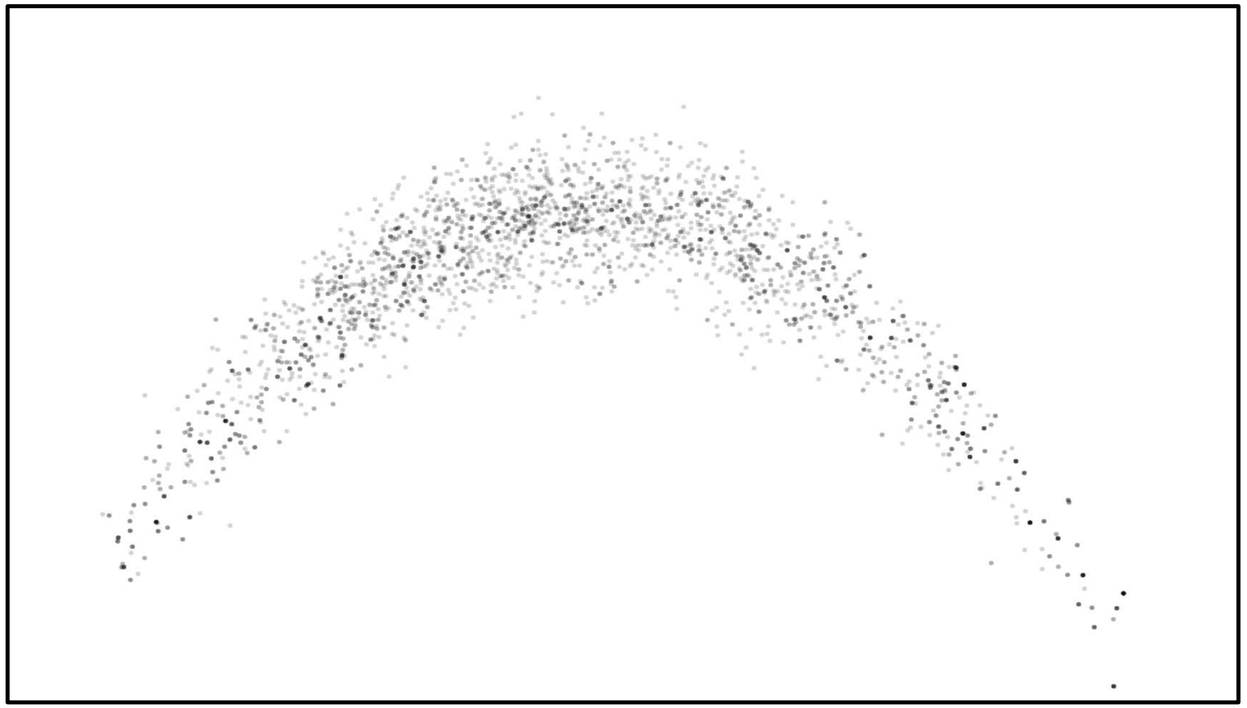
Gradient descent/ascent:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1} \nabla f(\boldsymbol{x}_{t-1})$$

scalar step size

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

Langevin MC:

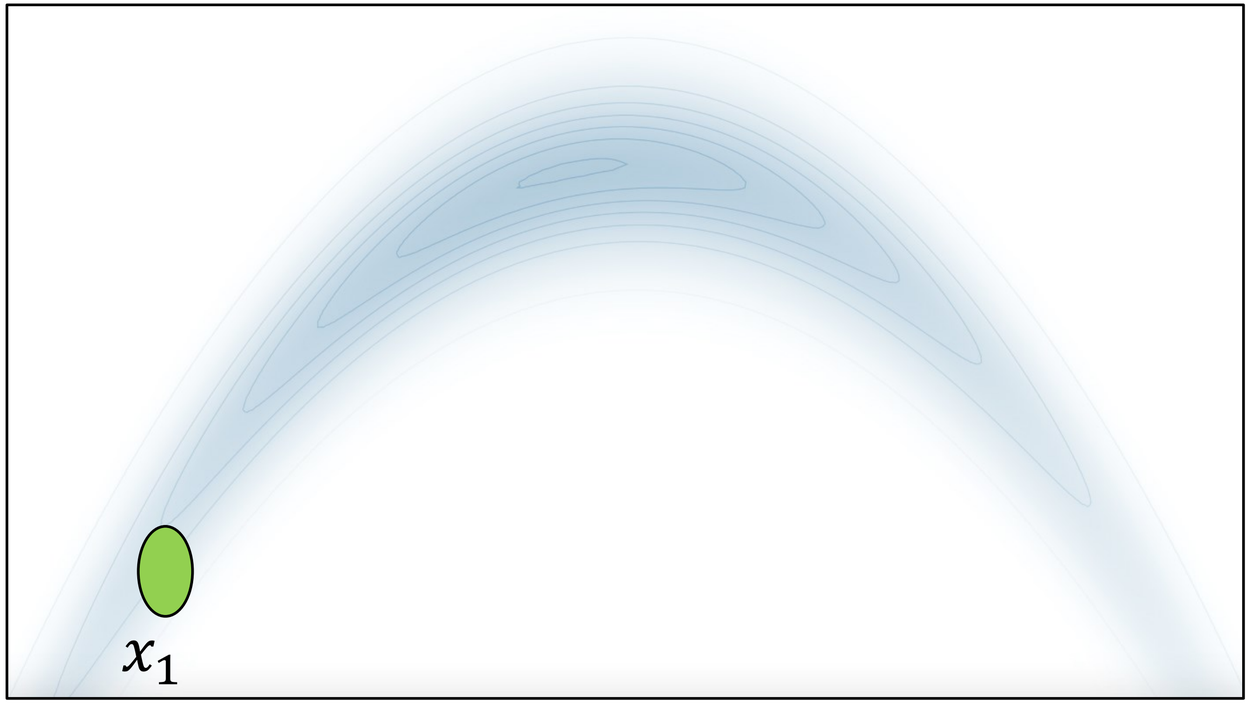
$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

Paul Langevin

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

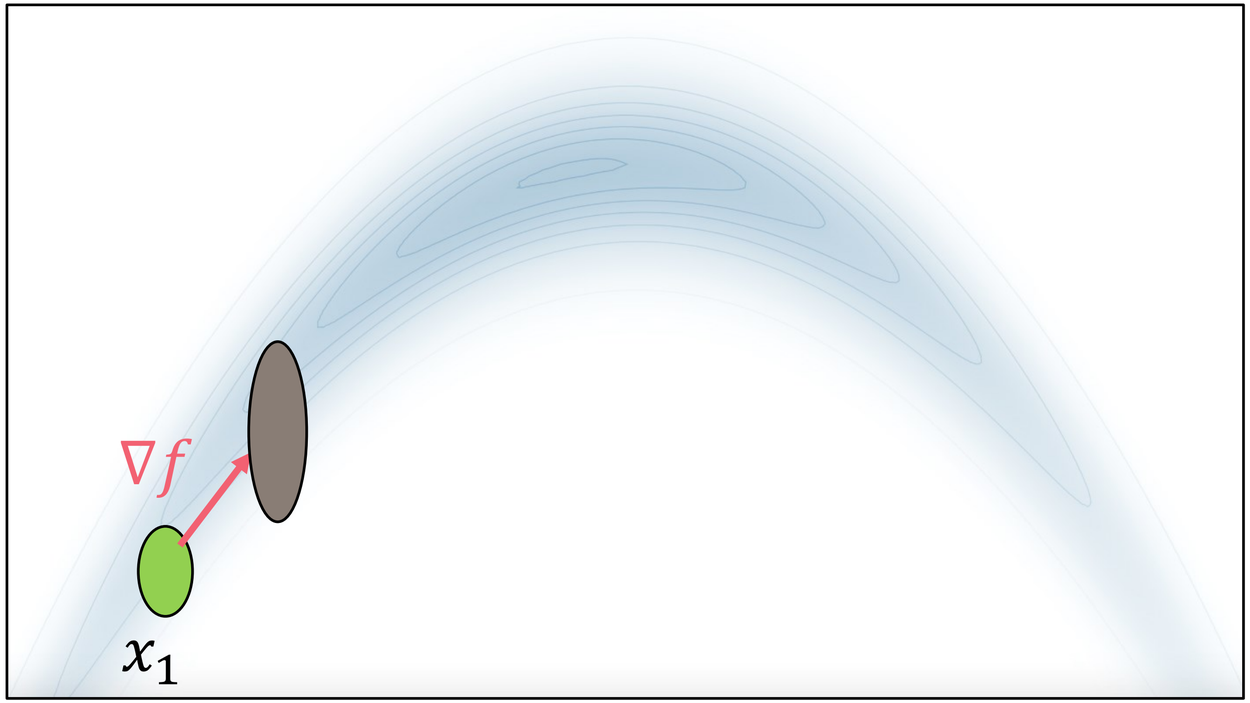
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

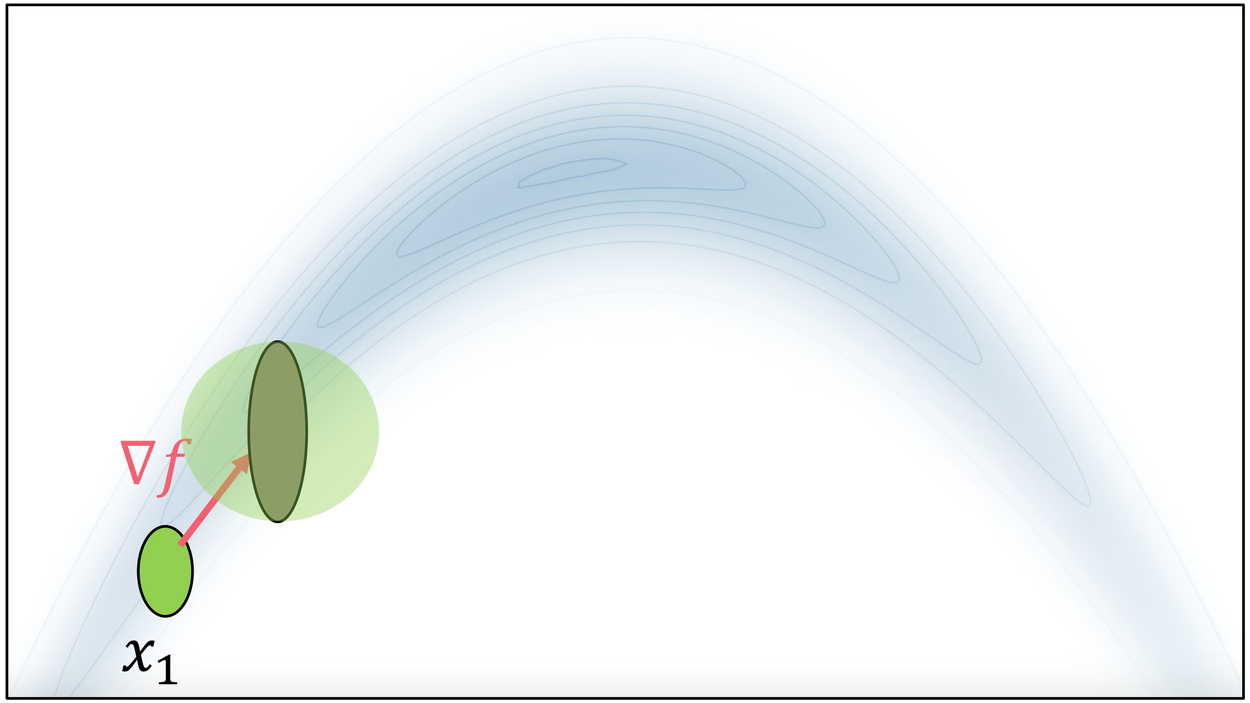
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

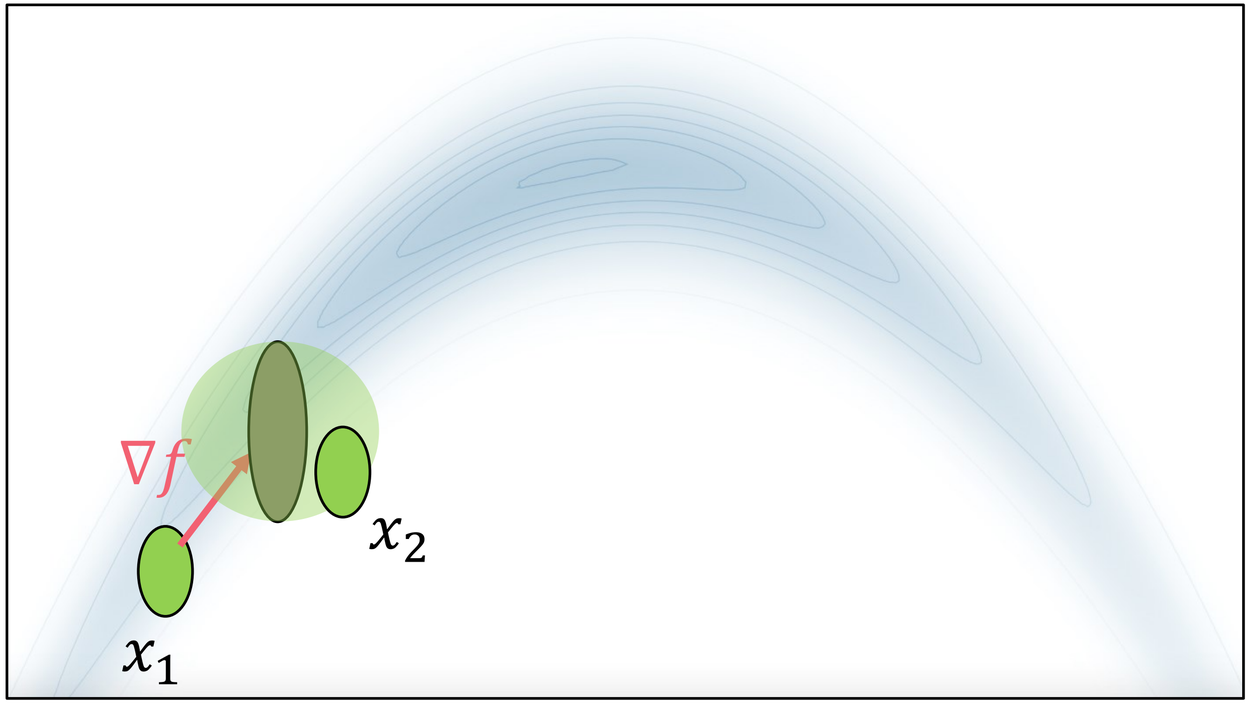
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

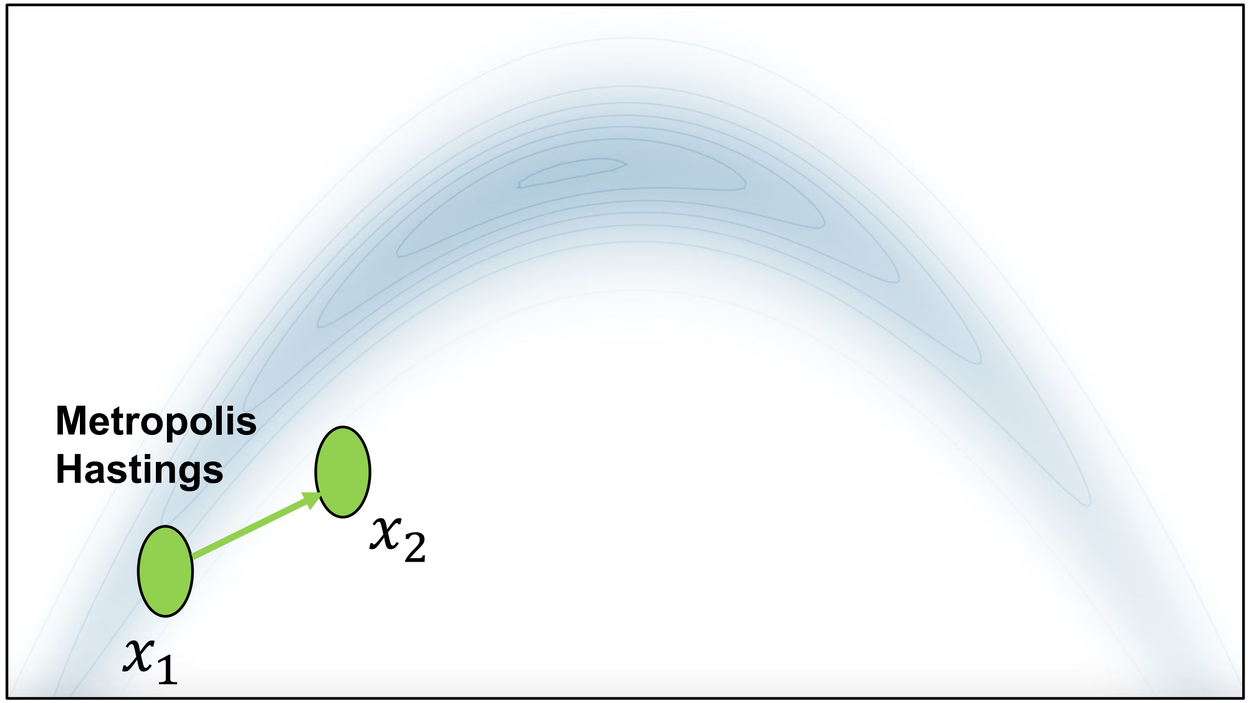
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

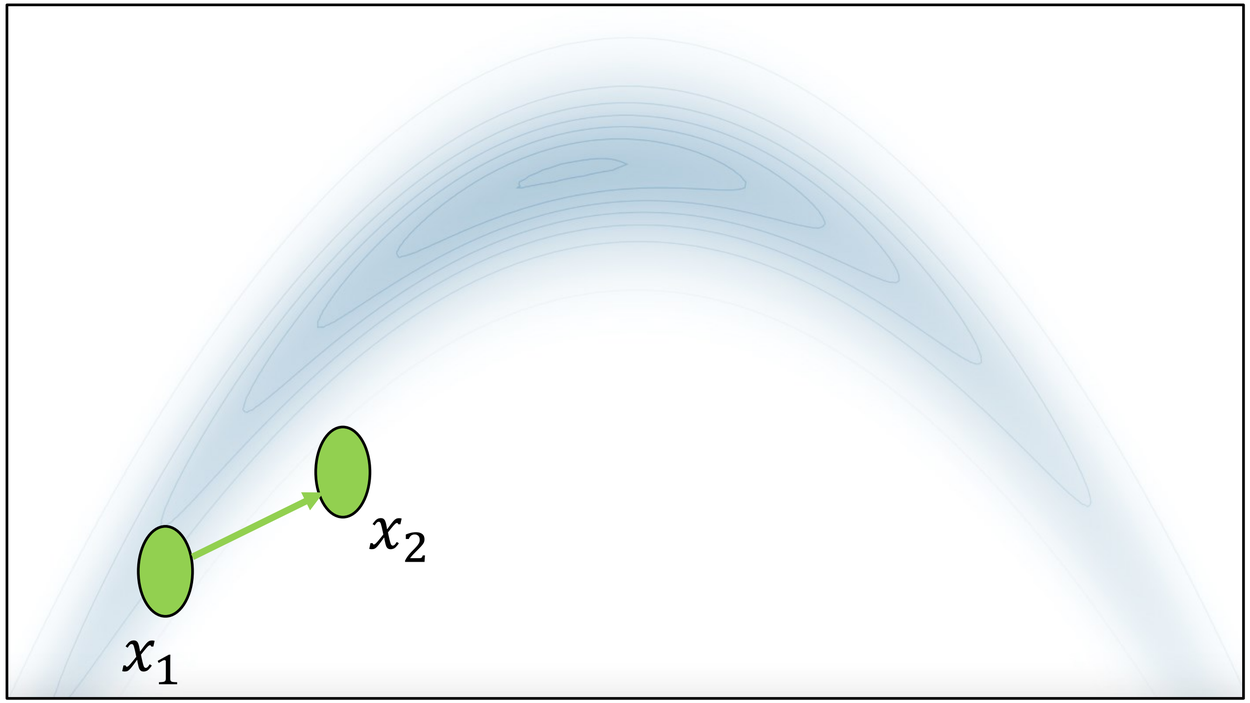
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

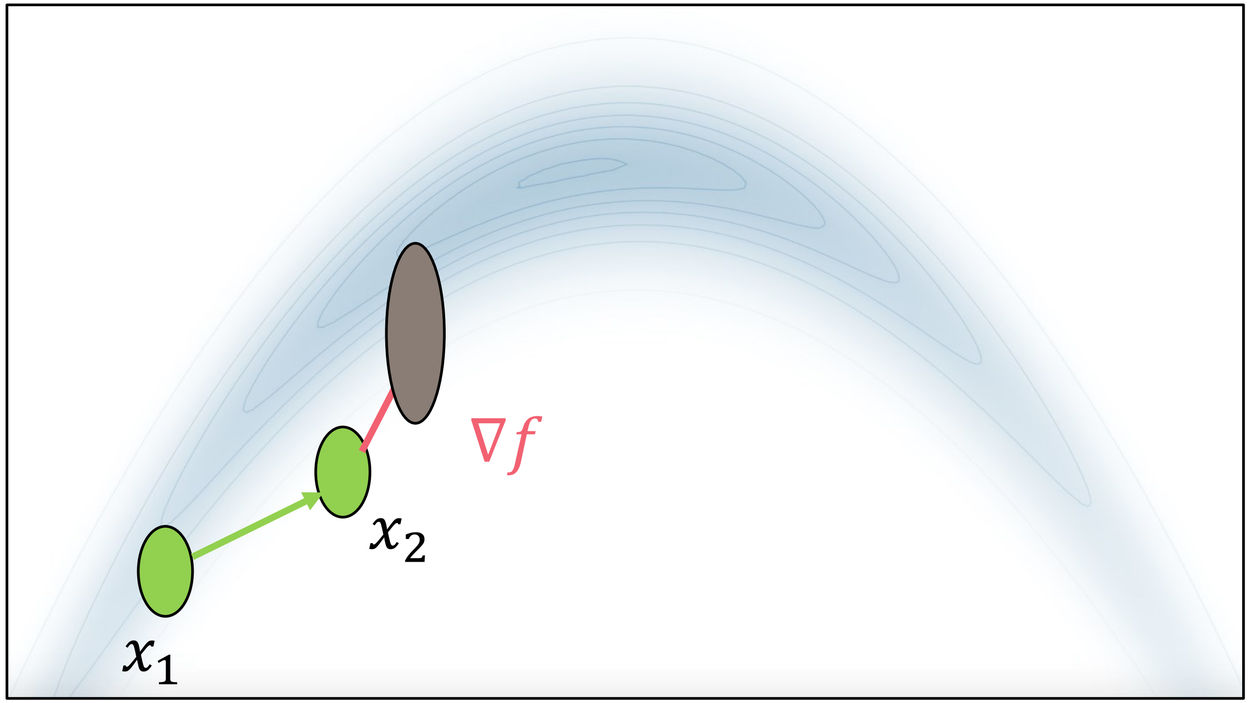
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

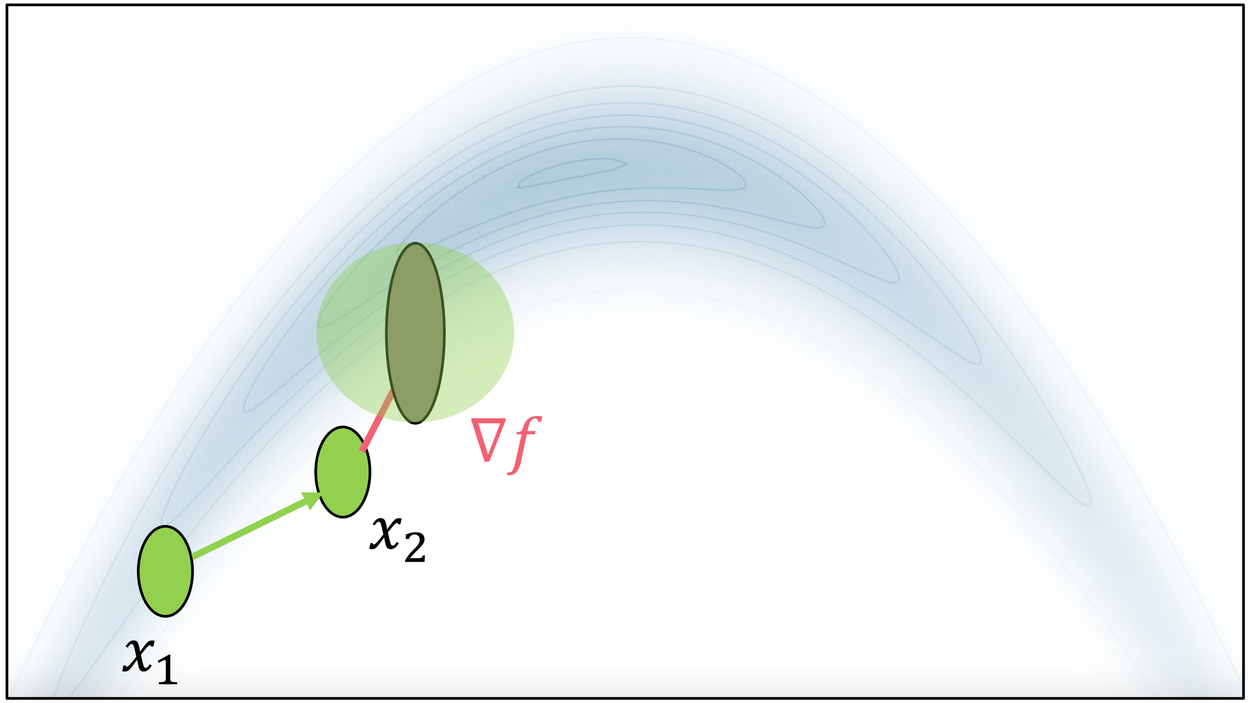
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

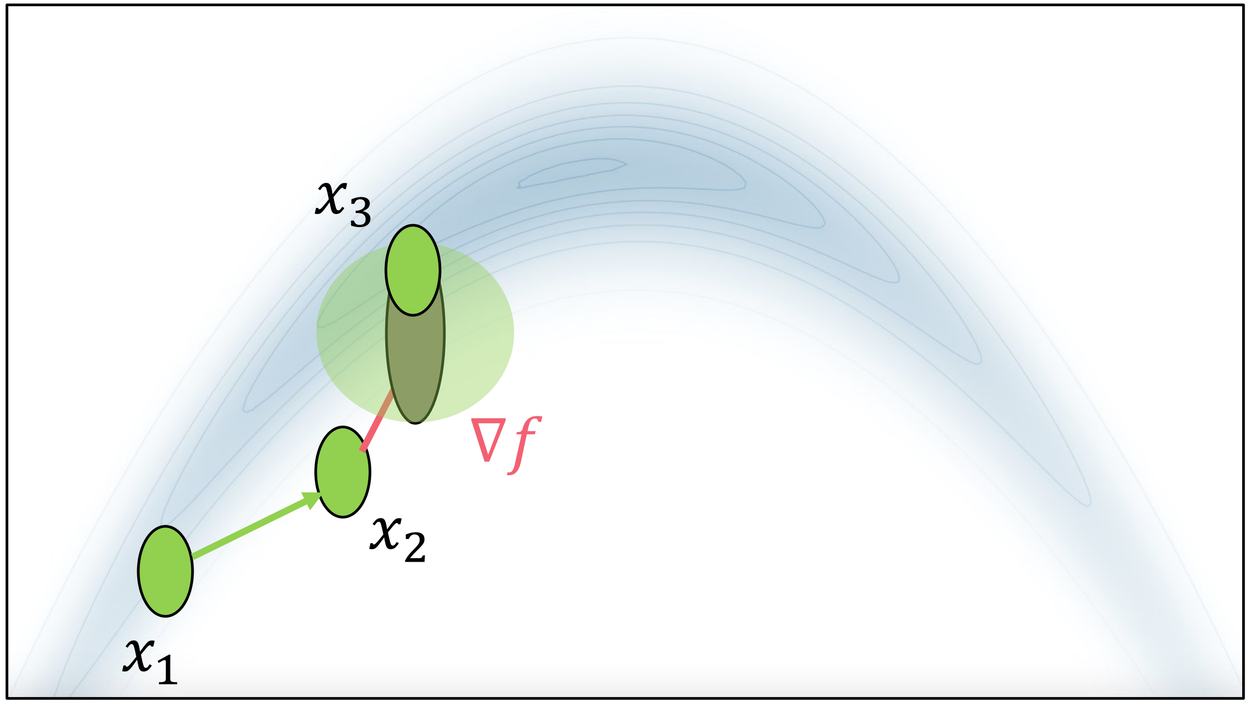
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

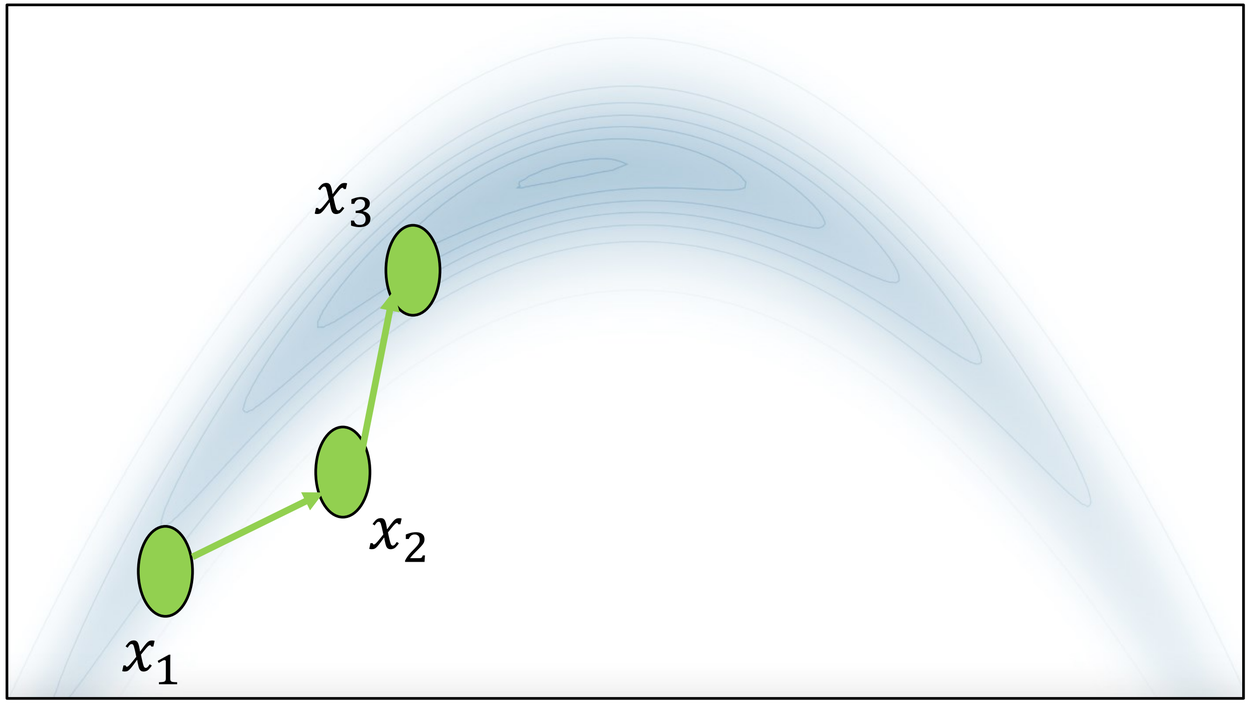
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

LMC OVERVIEW



Sampling problem:

$$\boldsymbol{x}_t \sim f$$

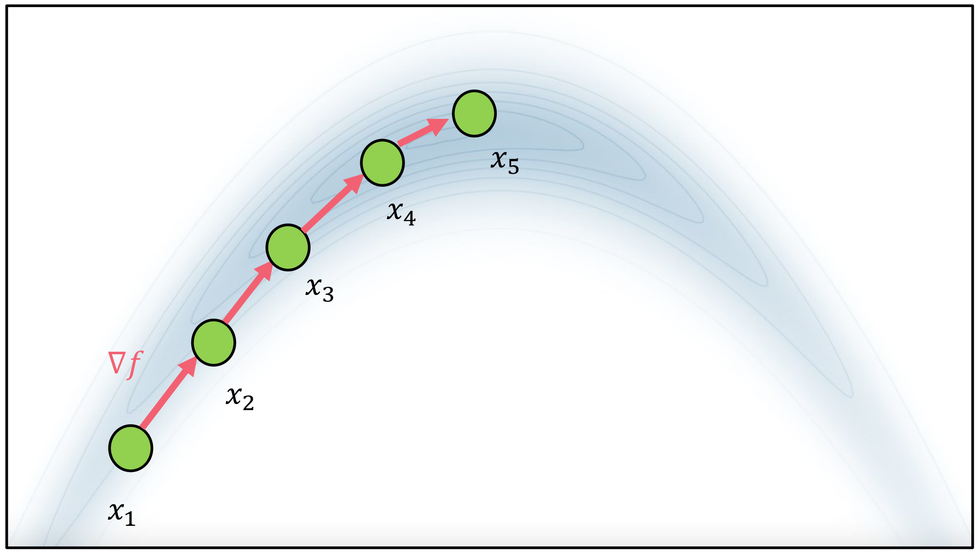
Langevin MC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

Apply Metropolis Hastings to accept/reject

[Luan et al., 2020]

GD VS. LMC

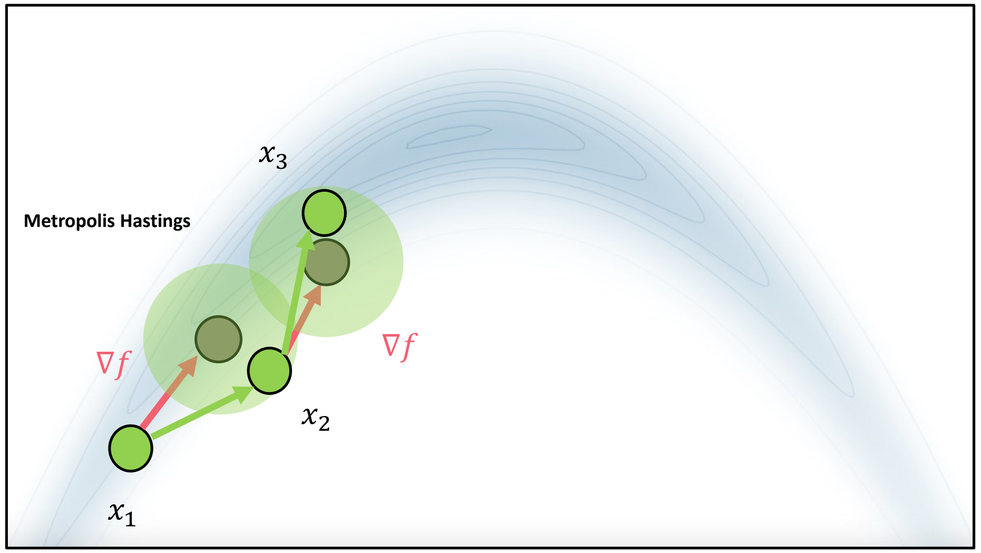


Optimization:

$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

GD:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1})$$



Sampling:

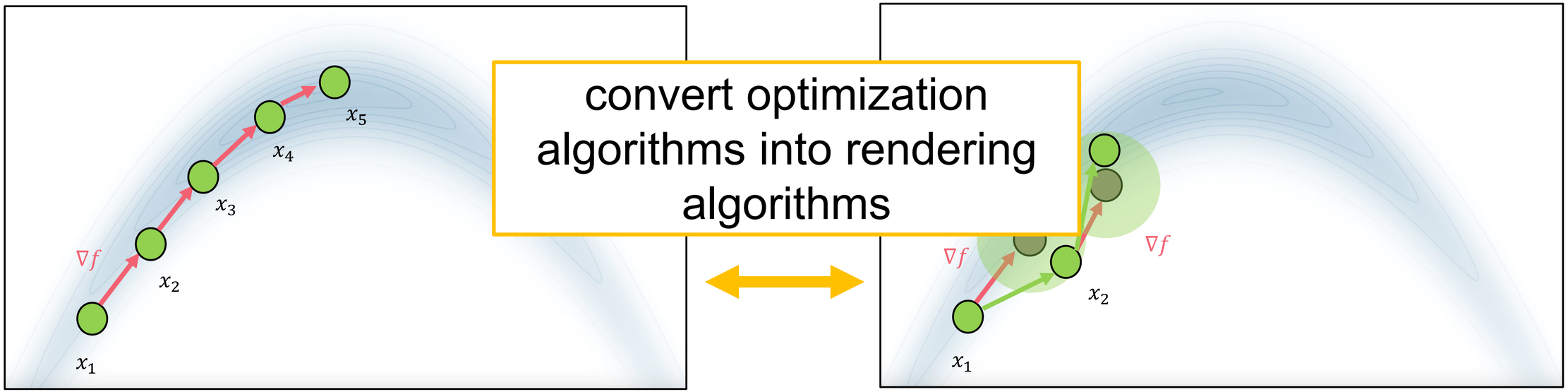
$$\boldsymbol{x}_t \sim f$$

LMC:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

[Luan et al., 2020]

GD VS. LMC



convert optimization algorithms into rendering algorithms

Optimization:

$$\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

GD:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1})$$

Sampling:

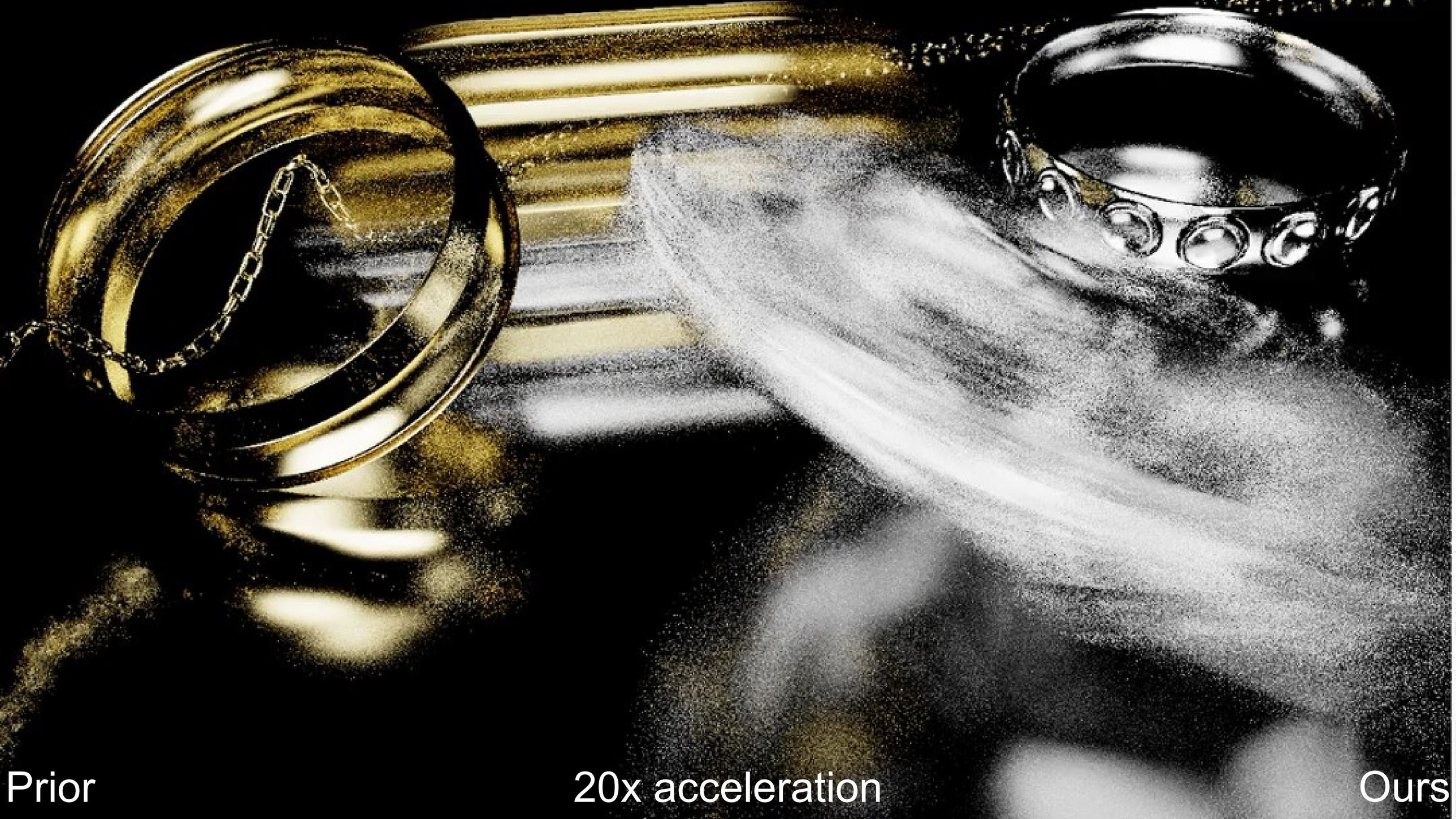
$$\boldsymbol{x}_t \sim f$$

LMC:

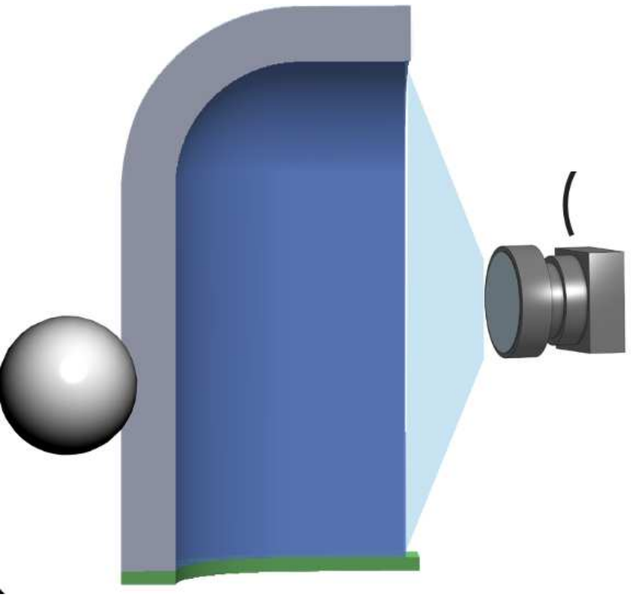
$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + s_{t-1}\nabla f(\boldsymbol{x}_{t-1}) + \frac{1}{s_{t-1}} N(0, \sigma^2 \boldsymbol{I})$$

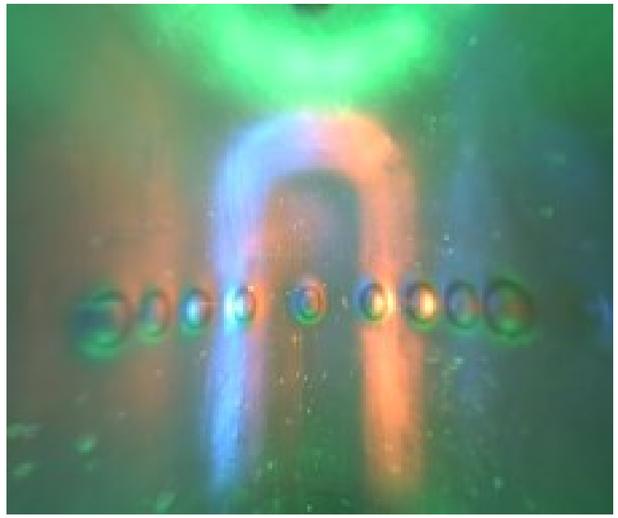
[Luan et al., 2020]

Prior 10x acceleration Ours


Prior
20x acceleration
Ours

Evaluating optical simulation framework

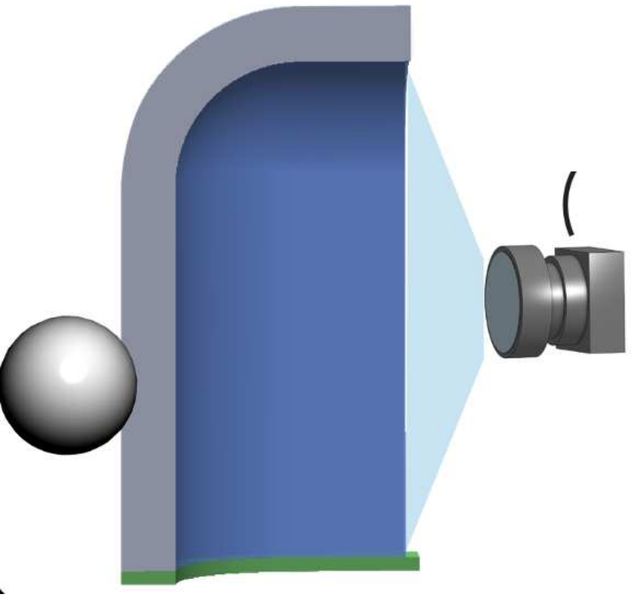


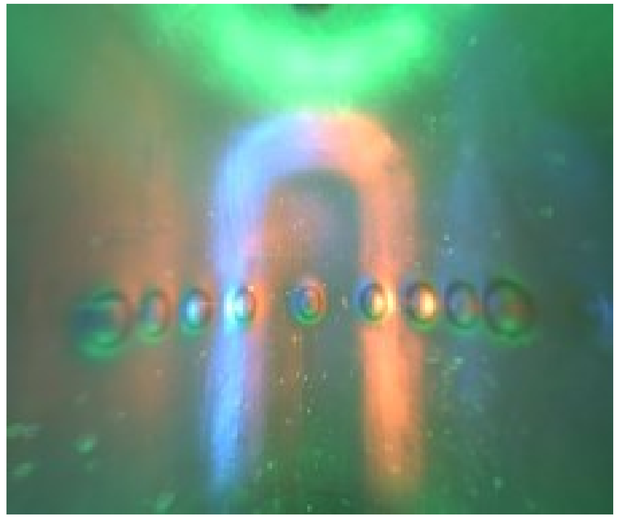


Real-world prototype

[Agarwal et al., 2023]

Evaluating optical simulation framework



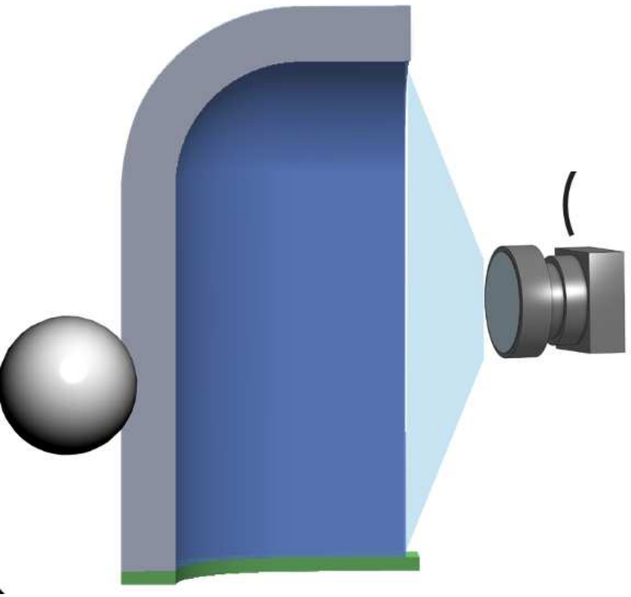


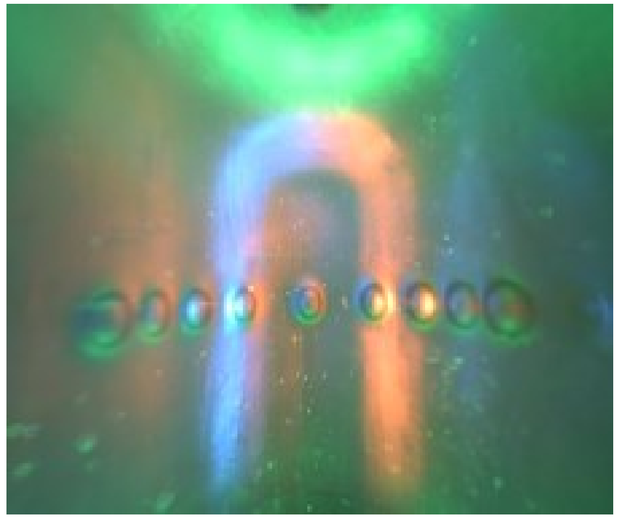
Real-world prototype

Rasterization

[Agarwal et al., 2023]

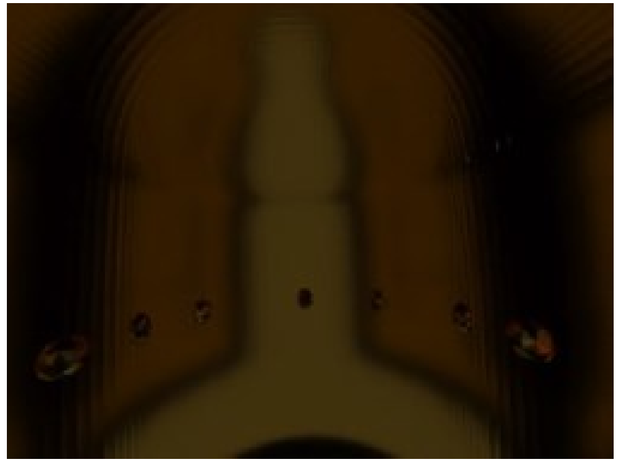
Evaluating optical simulation framework





Real-world prototype

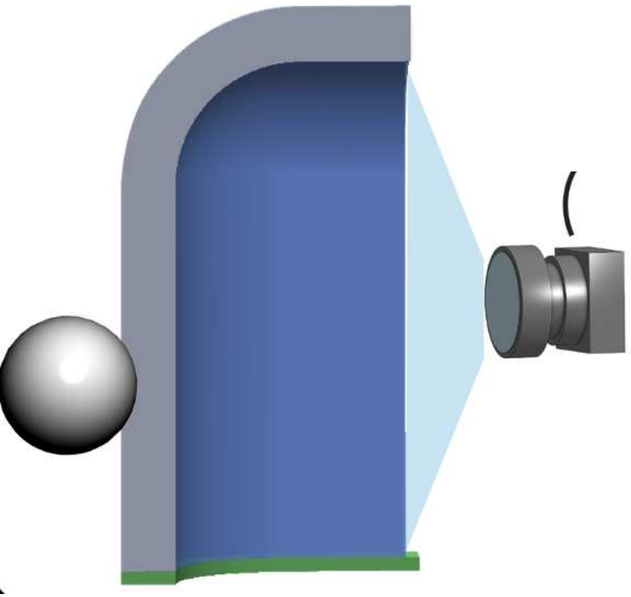
Rasterization

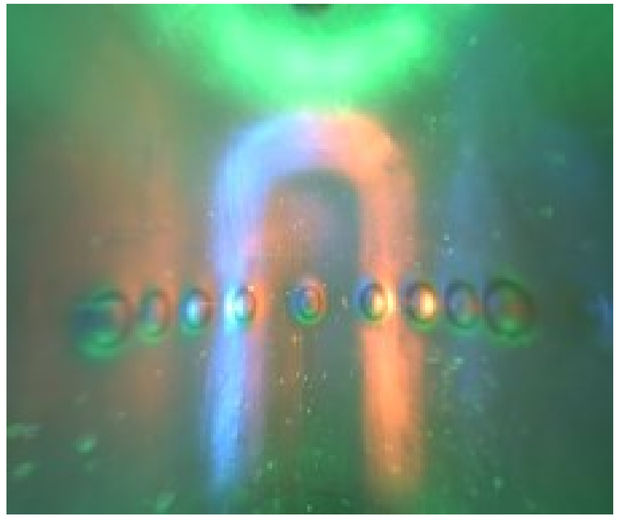


Path tracing

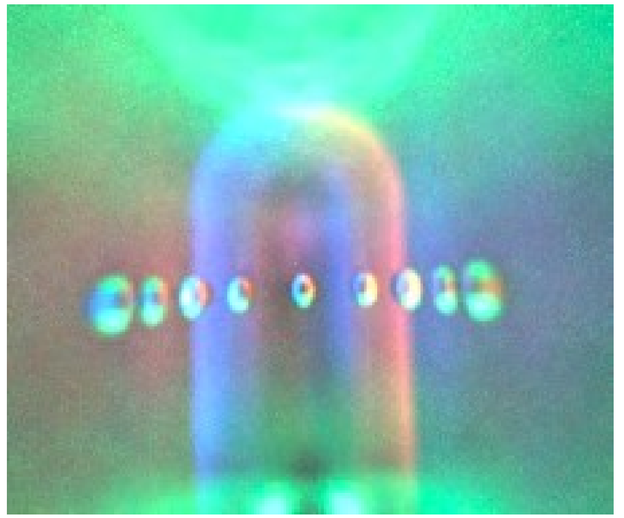
[Agarwal et al., 2023]

Evaluating optical simulation framework





Real-world prototype



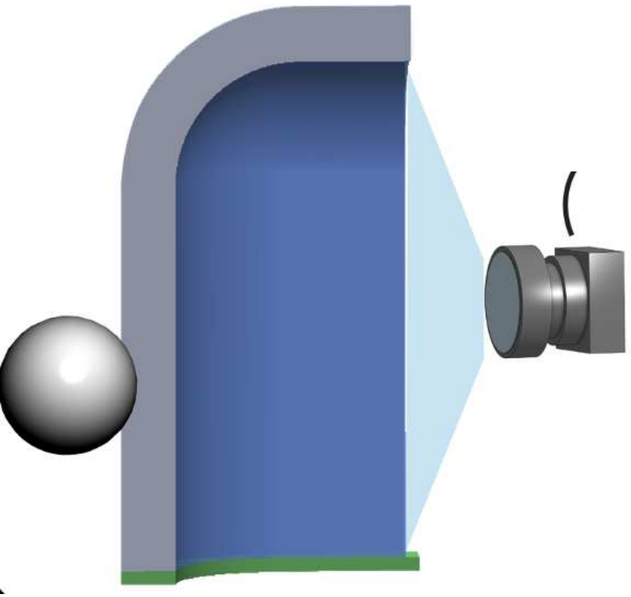
Our simulation result

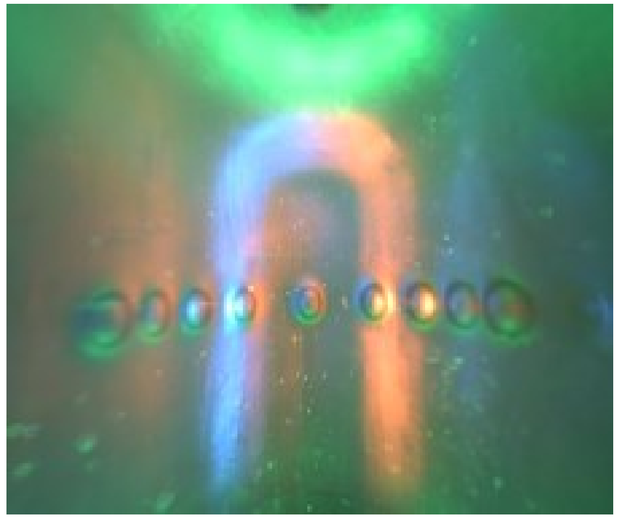
Rasterization

Path tracing

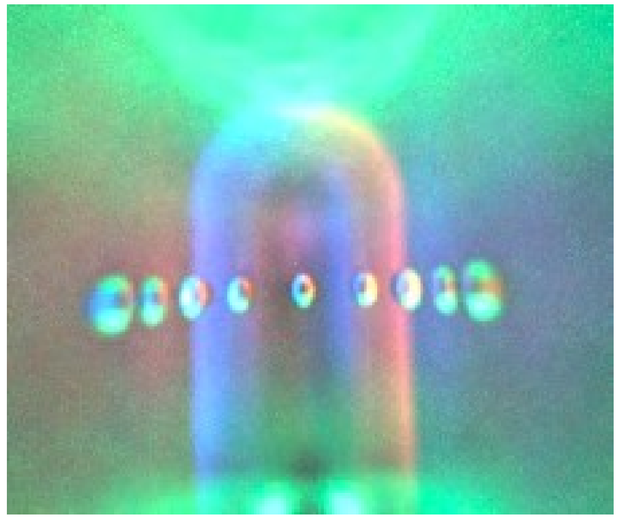
[Agarwal et al., 2023]

Evaluating optical simulation framework





Real-world prototype



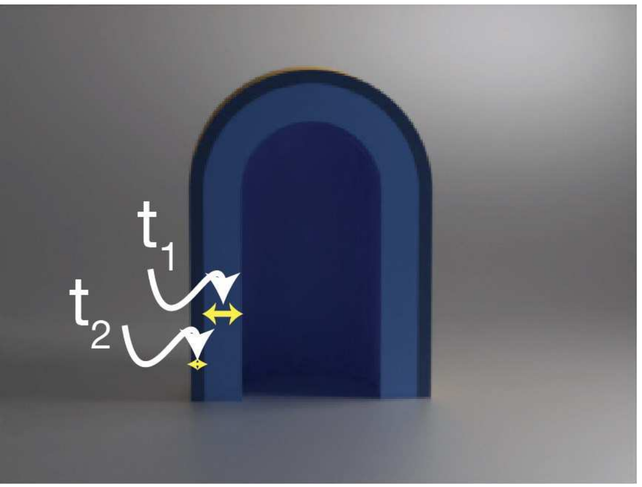
Our simulation result

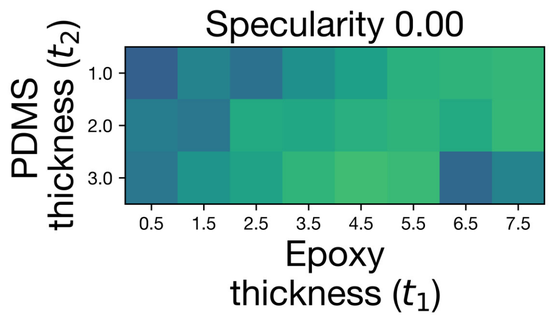
Rasterization

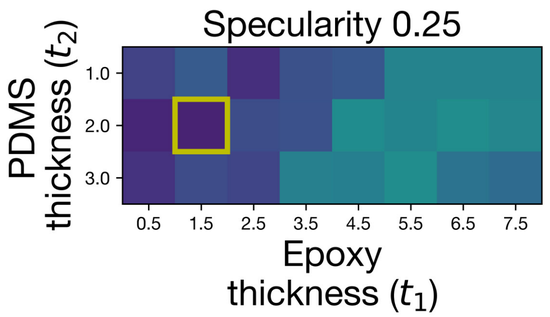
Path tracing

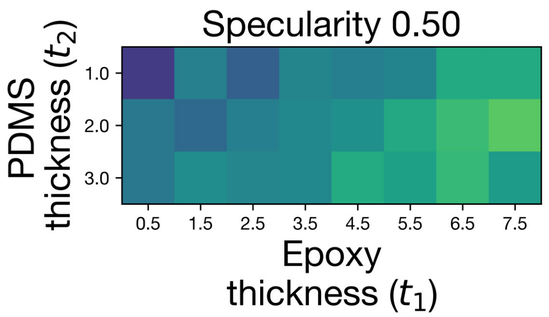
[Agarwal et al., 2023]

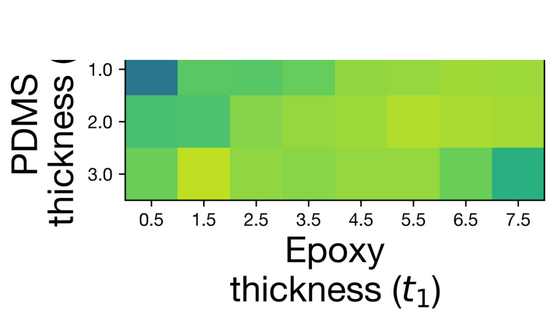
Curved sensor material design

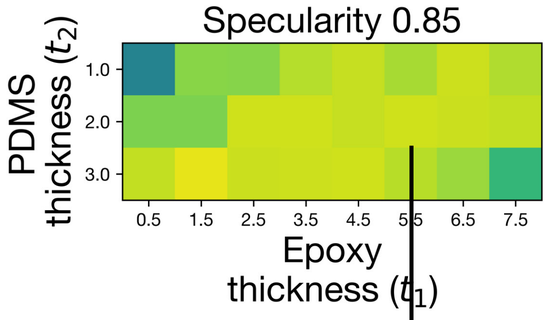


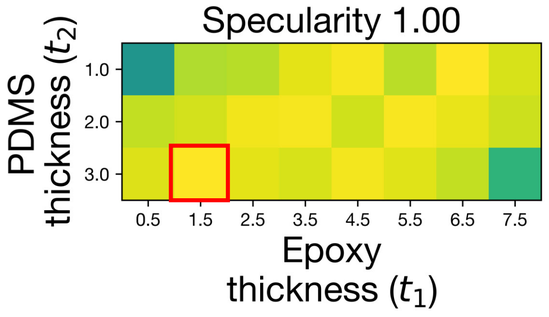


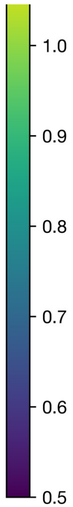


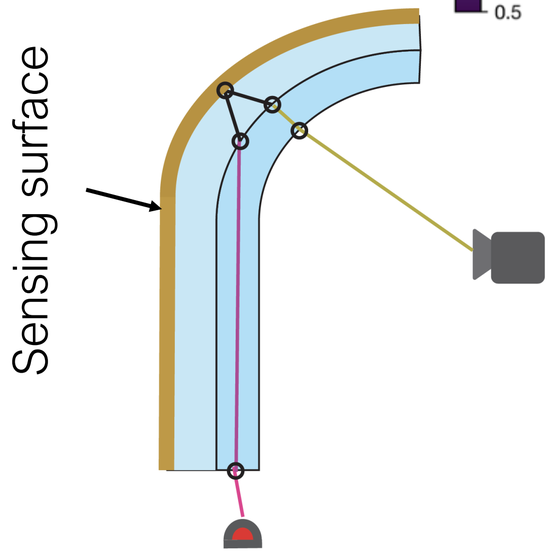


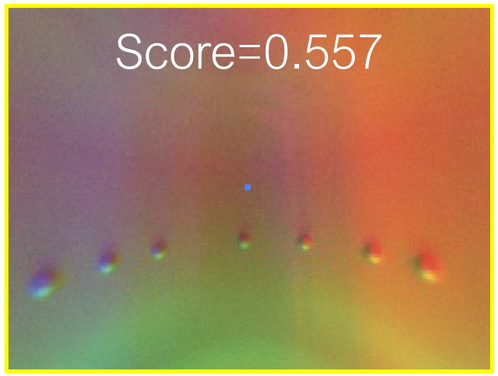


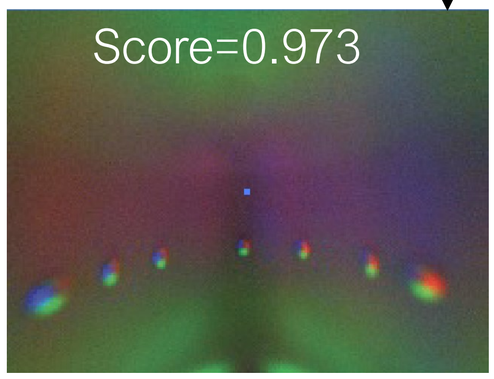






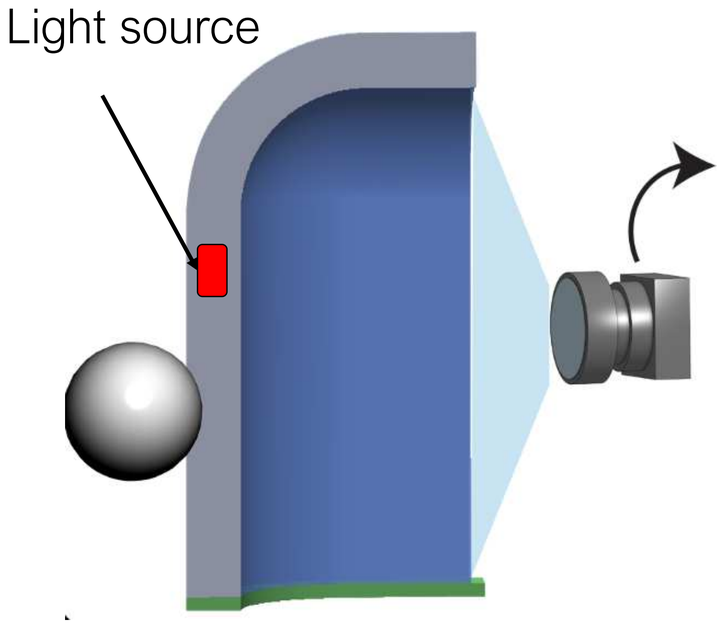


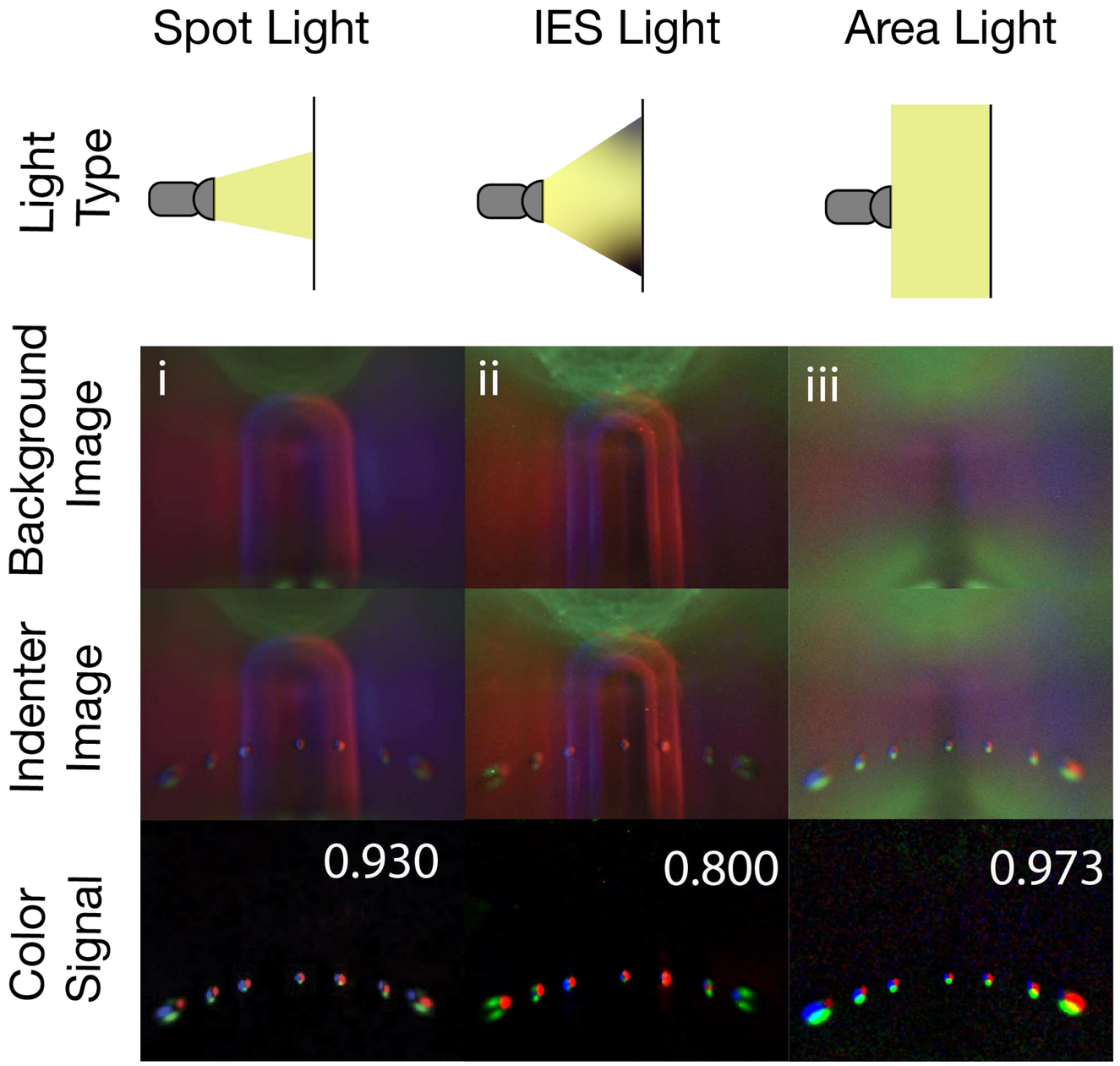




[Agarwal et al., 2023]

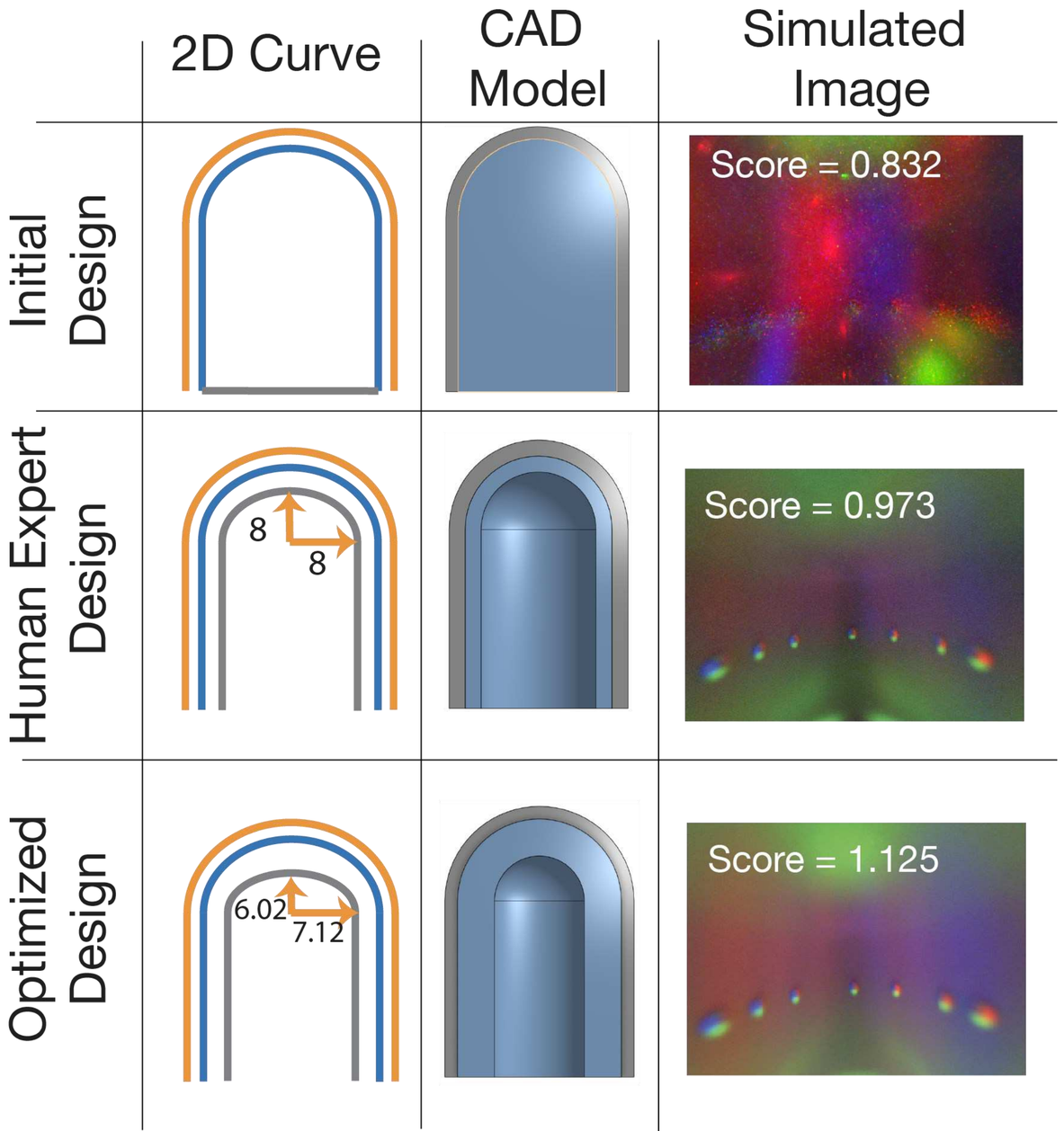
Curved sensor illumination design





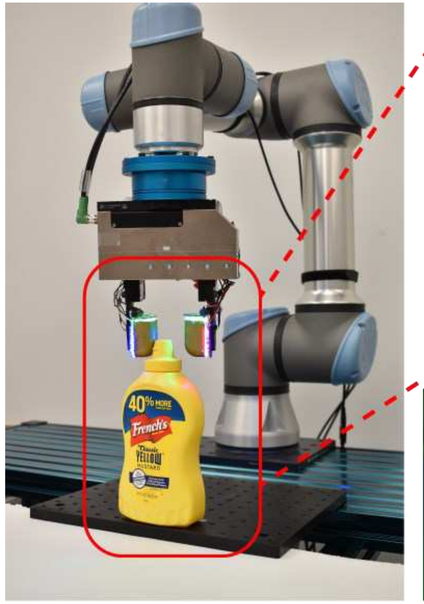
Curved sensor shape design

- Used gradient-free optimization, CMA-ES, for optimizing the sensor shape
- Optimized sensor design is 35% better than initial design
- Optimized sensor design outperforms human-expert design in 3D shape reconstruction



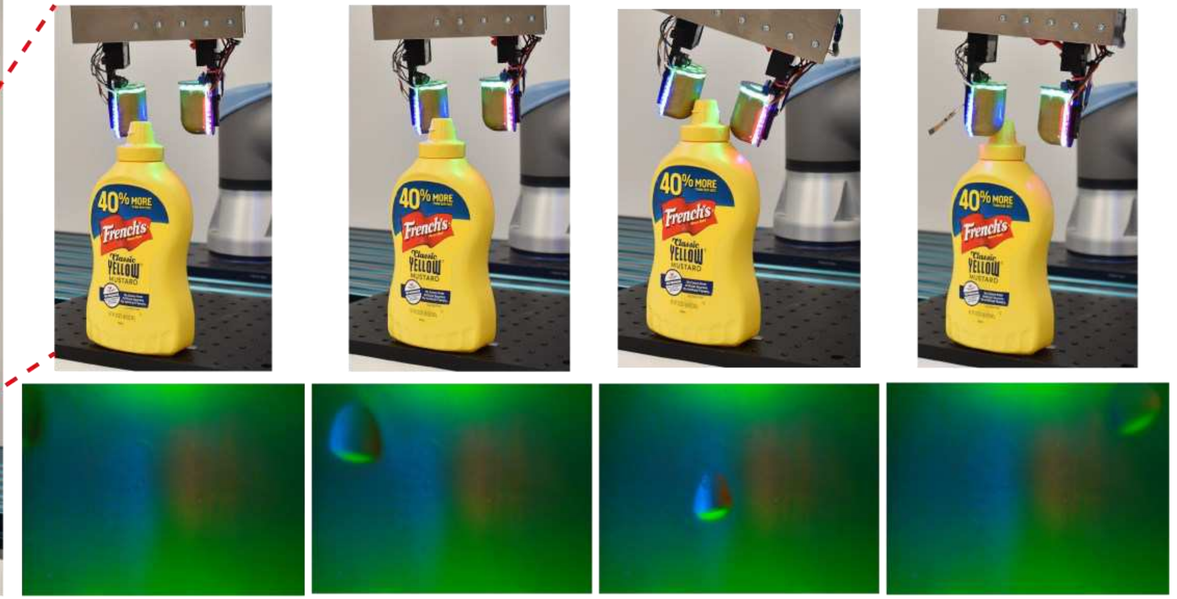
Sensing Surface
Interface between shell and elastomer
Shell Surface
Hard shell
Soft elastomer

Results: Robotic grasping

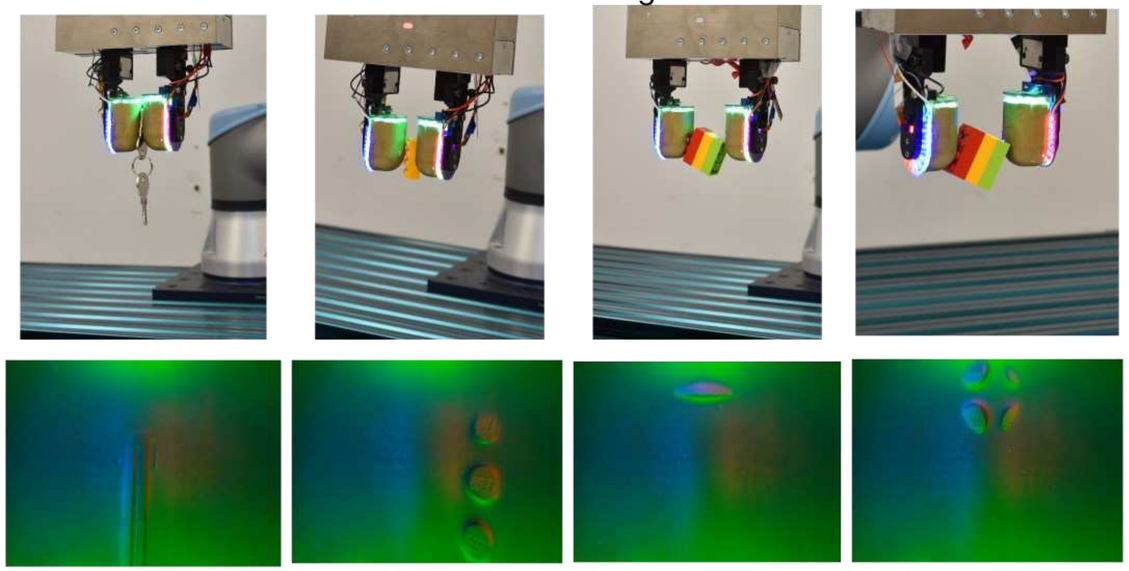


Robotic arm with optimized tactile sensor

Common Objects



Tactile Images

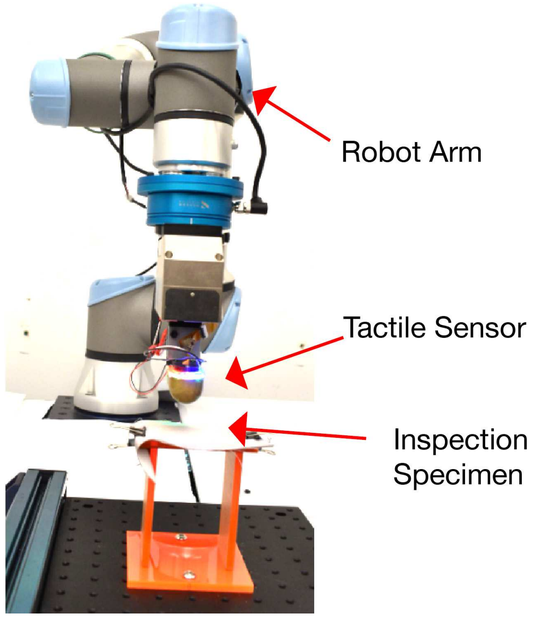


Tactile Images

[Agarwal et al., 2023]

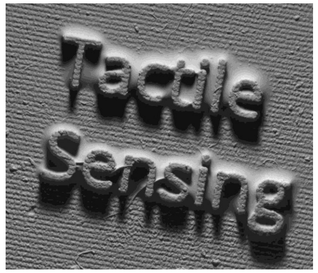
Results: surface inspection

Experiment 1: Detection accuracy vs text size



Experiment setup

Text description:
Tactile
Sensing

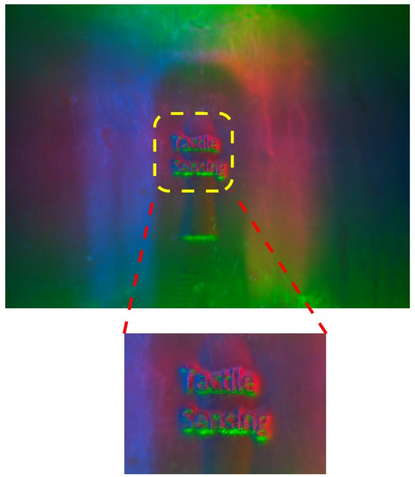


Text size = 1.5 mm

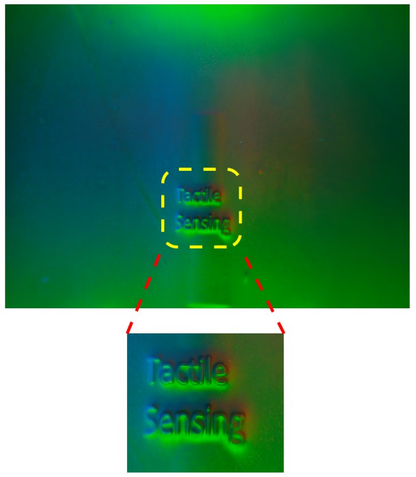
Text size = 1.0 mm

Human expert design

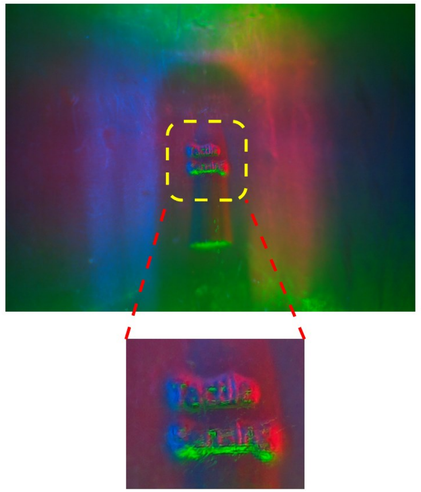
Optimized design



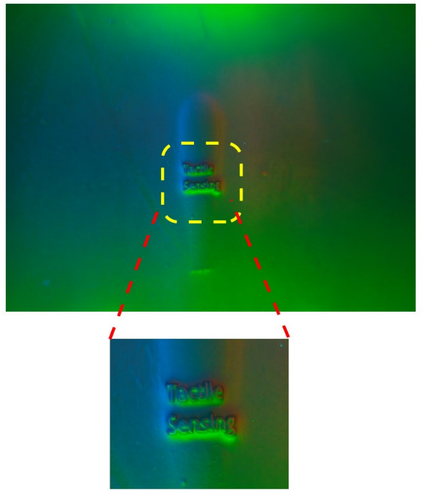
“Tatle Sending”



“Tacolle Sending”



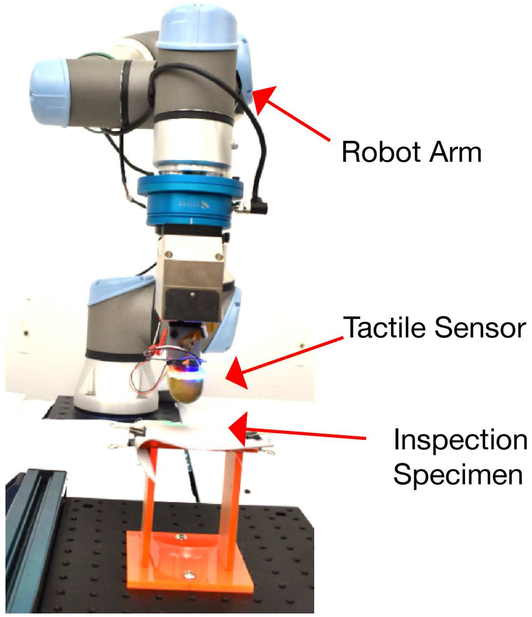
“_”



“Tacolle Sensing”

[Agarwal et al., 2023]

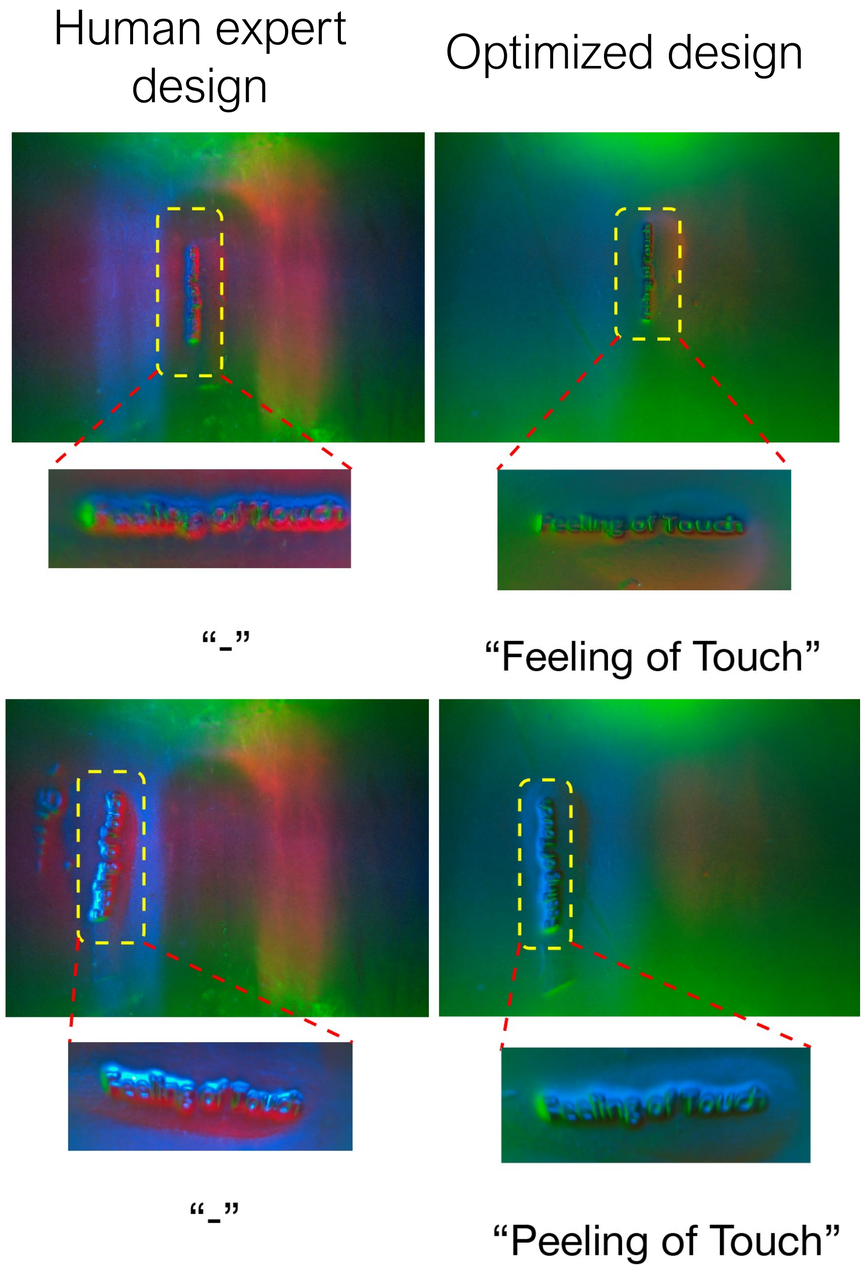
Results: surface inspection



Experiment setup

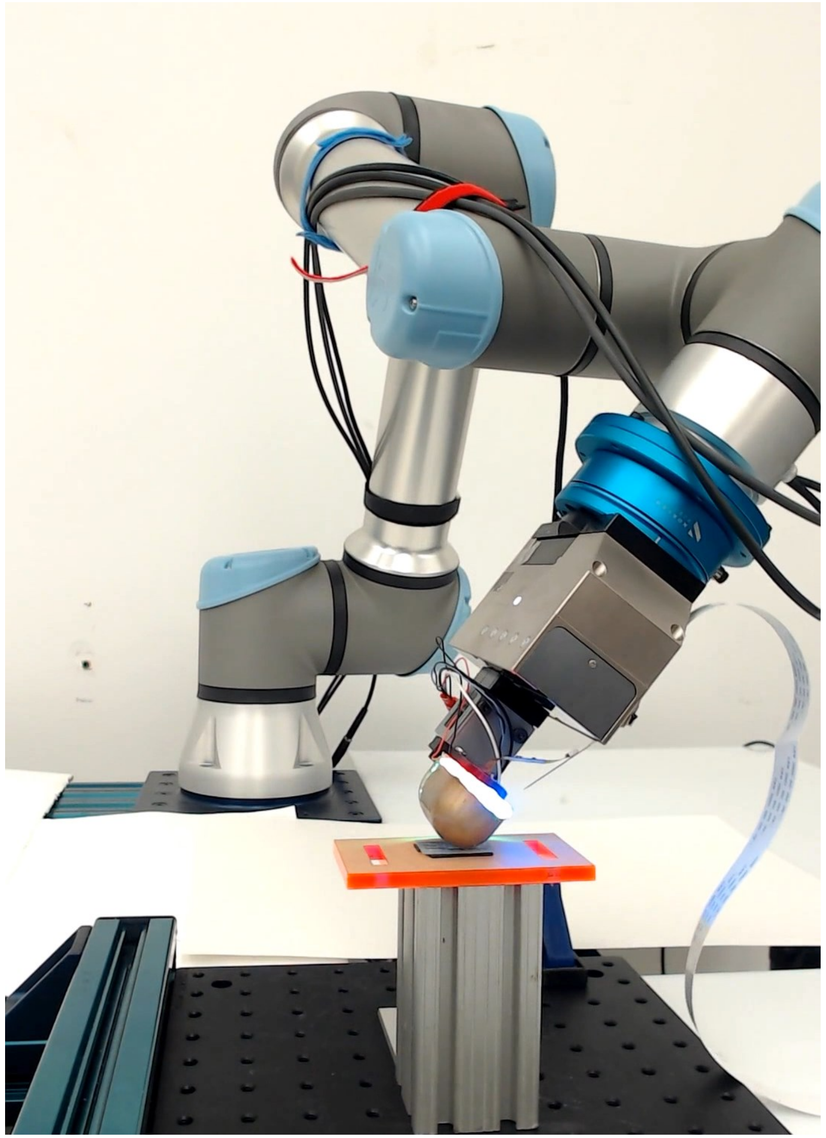
Text description: Feeling of Touch

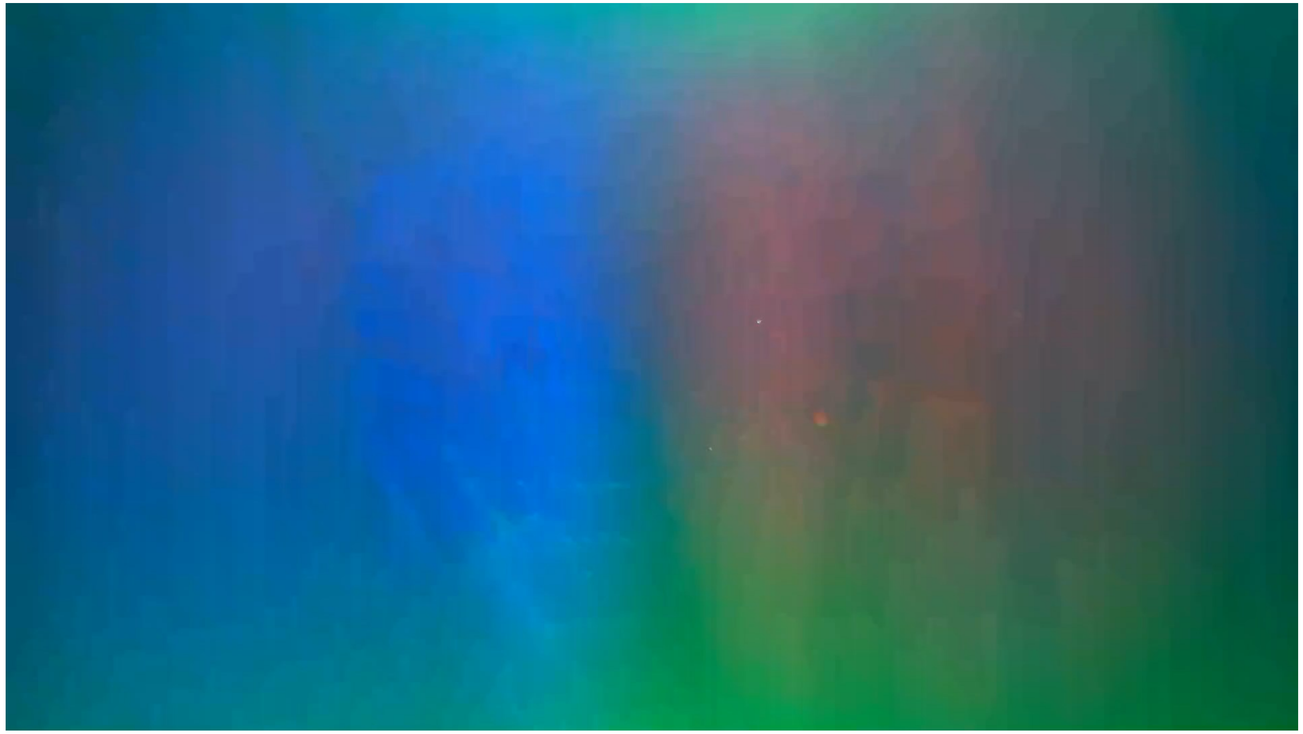
Experiment 2: Detection accuracy vs contact location



[Agarwal et al., 2023]

Results: surface inspection





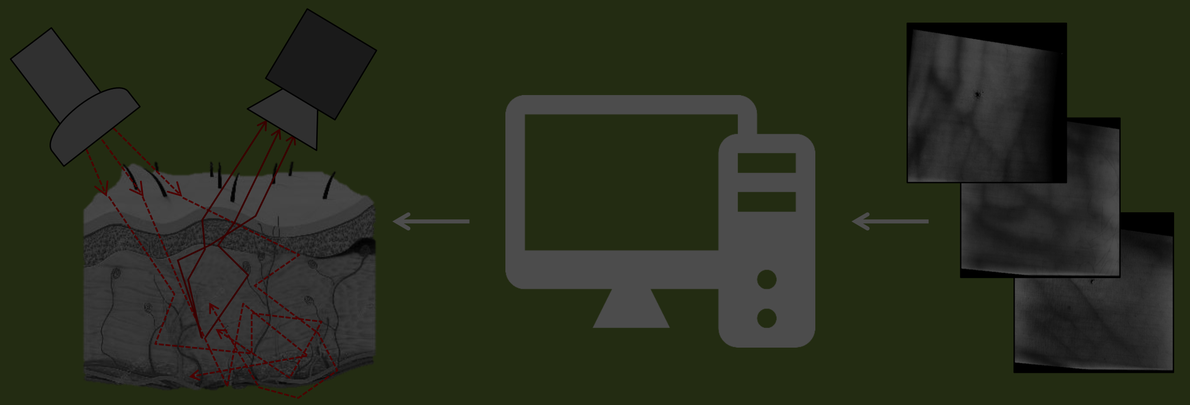
[Agarwal et al., 2023]

Physics-based rendering and its applications to computational imaging

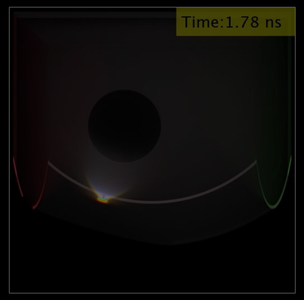
forward rendering

- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

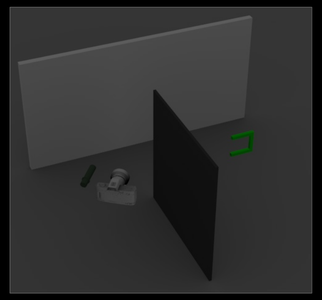
inverse rendering



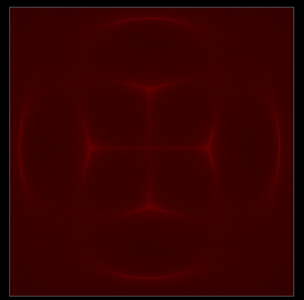
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



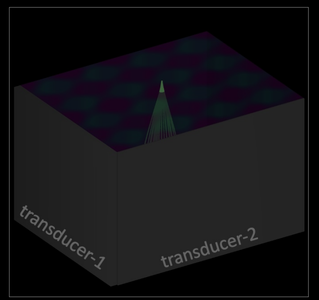
time-of-flight imaging



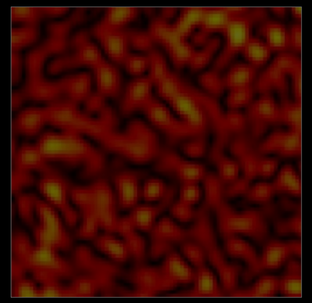
non-line-of-sight imaging



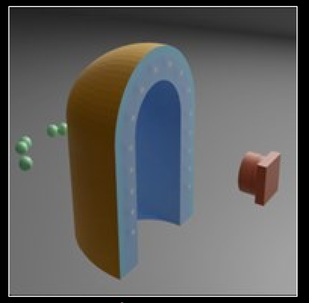
acousto-optic lensing



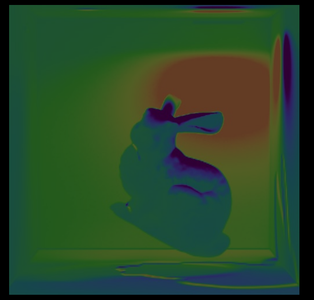
ultrafast light scanning



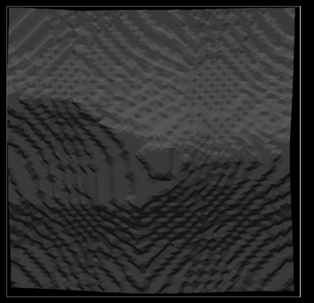
speckle imaging



tactile sensor design



differentiable renderer



inverse problems

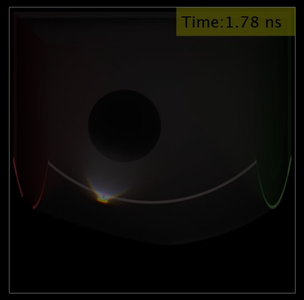
Physics-based rendering and its applications to computational imaging

forward rendering

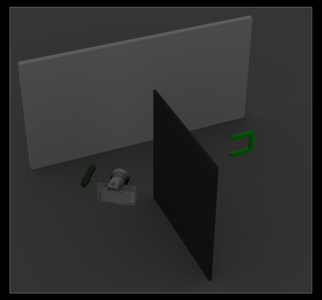
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

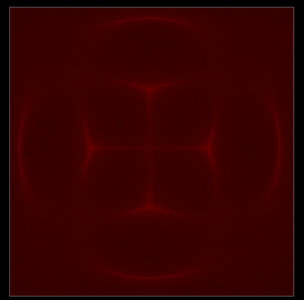
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



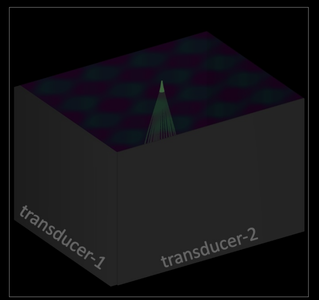
time-of-flight imaging



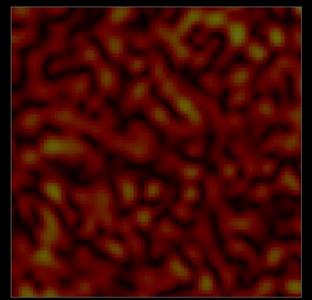
non-line-of-sight imaging



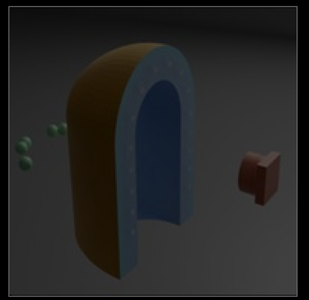
acousto-optic lensing



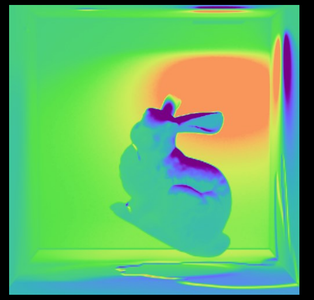
ultrafast light scanning



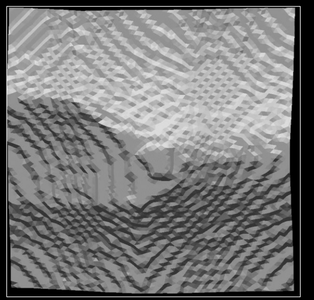
speckle imaging



tactile sensor design

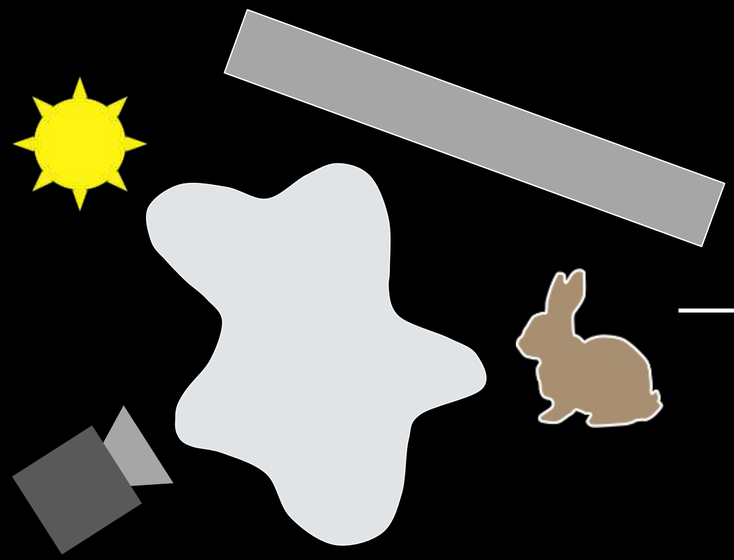


differentiable renderer



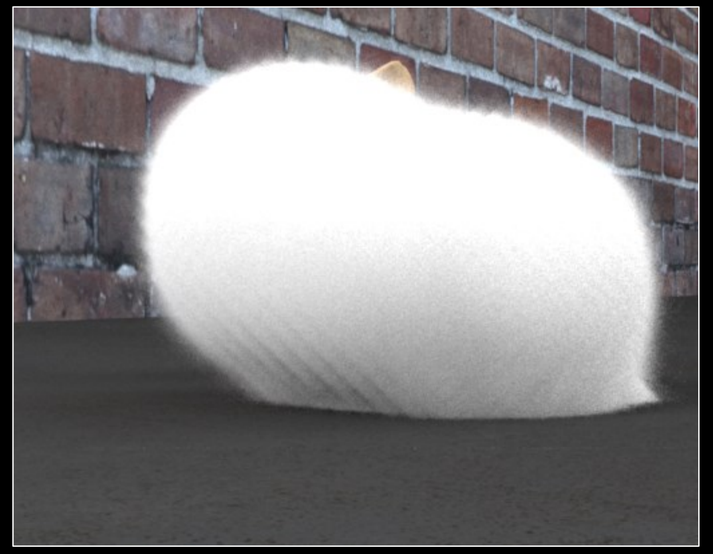
inverse problems

Forward rendering



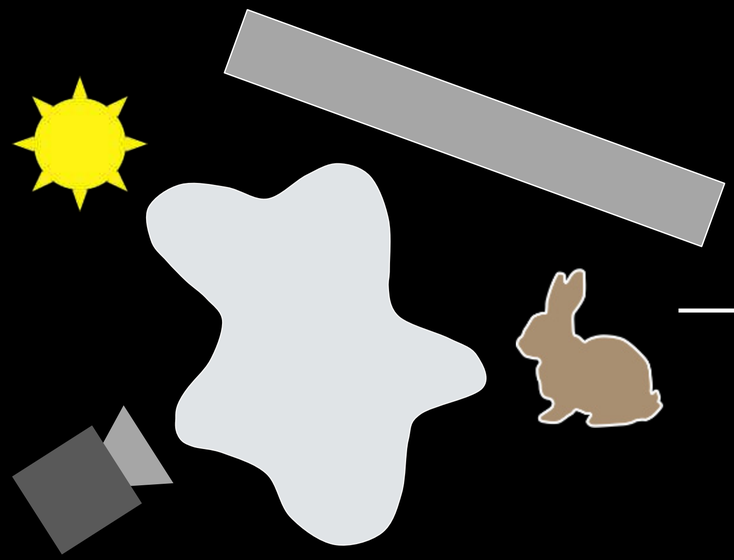
digital scene specification
(geometry, materials,
optics, light sources)

physically-accurate
rendering



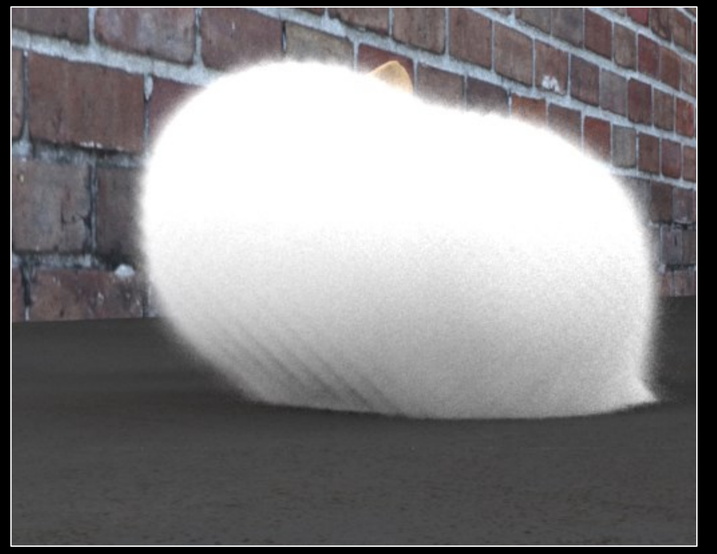
photorealistic
simulated image

Inverse rendering



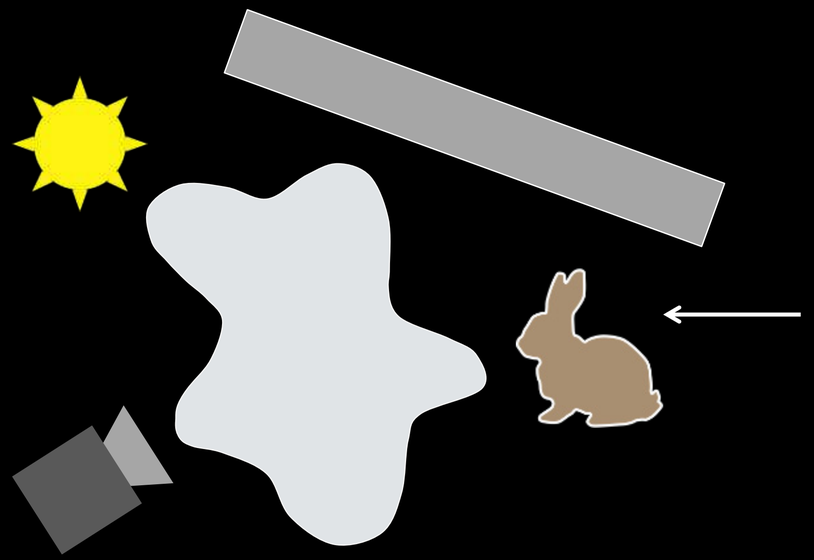
digital scene specification
(geometry, materials,
camera, light sources)

physically-accurate
inverse rendering



photorealistic
synthetic image

Inverse rendering



digital scene specification
(geometry, materials,
camera, light sources)

physically-accurate
inverse rendering

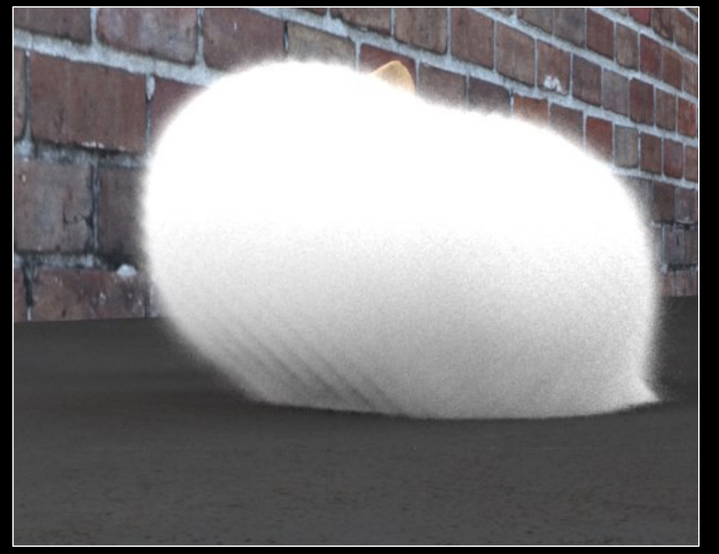
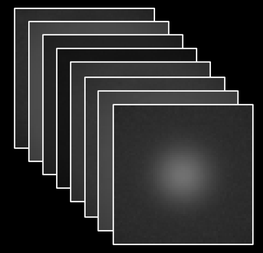


image measurements

Analysis by synthesis (a.k.a. inverse rendering)

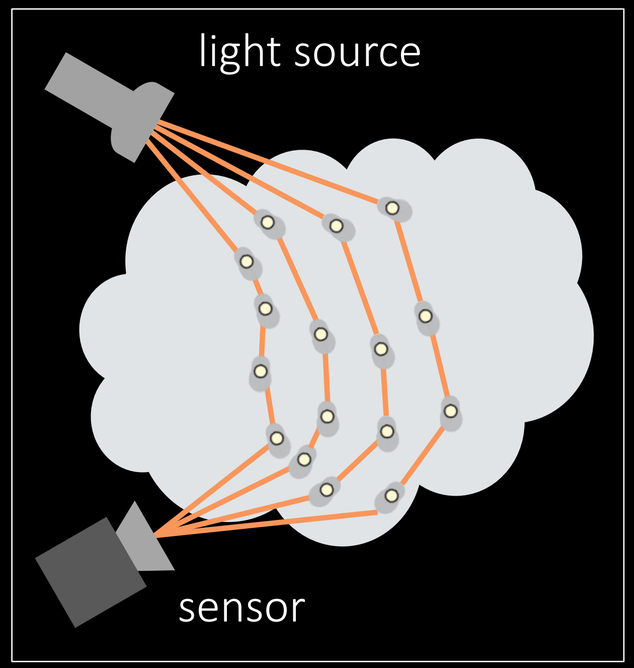
$$\min_{\text{unknowns m (tissue properties)}} \left\| \quad - \; \text{image}(m) \right\|^2$$



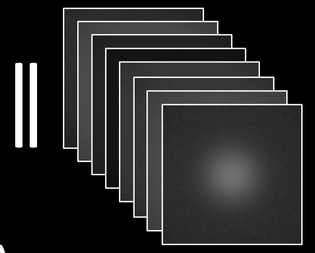
Analysis by synthesis (a.k.a. inverse rendering)

captured measurements

$$\min_{\text{unknowns } m \text{ (tissue properties)}} \left\| \quad - \text{image}(m) \right\|^2$$



Analysis by synthesis (a.k.a. inverse rendering)

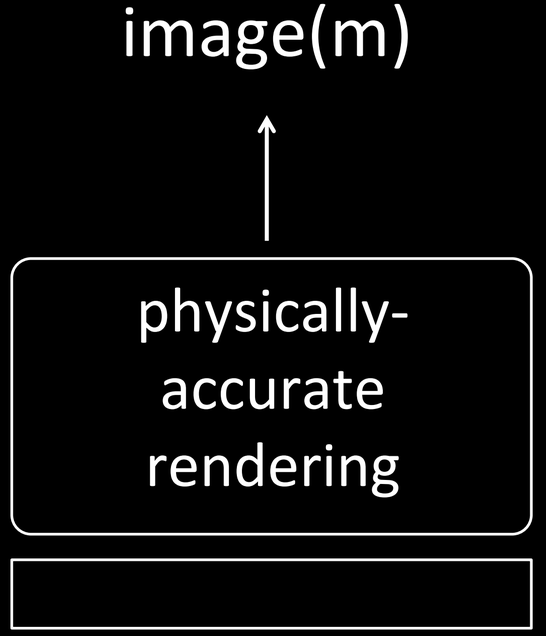
$$\min_{\text{unknowns m (tissue properties)}} \left\| \text{  } - \text{image(m)} \right\|^2$$

solve with exhaustive search

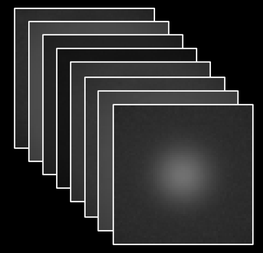
for all parameters m

evaluate loss(m)

computed with rendering



Analysis by synthesis (a.k.a. inverse rendering)

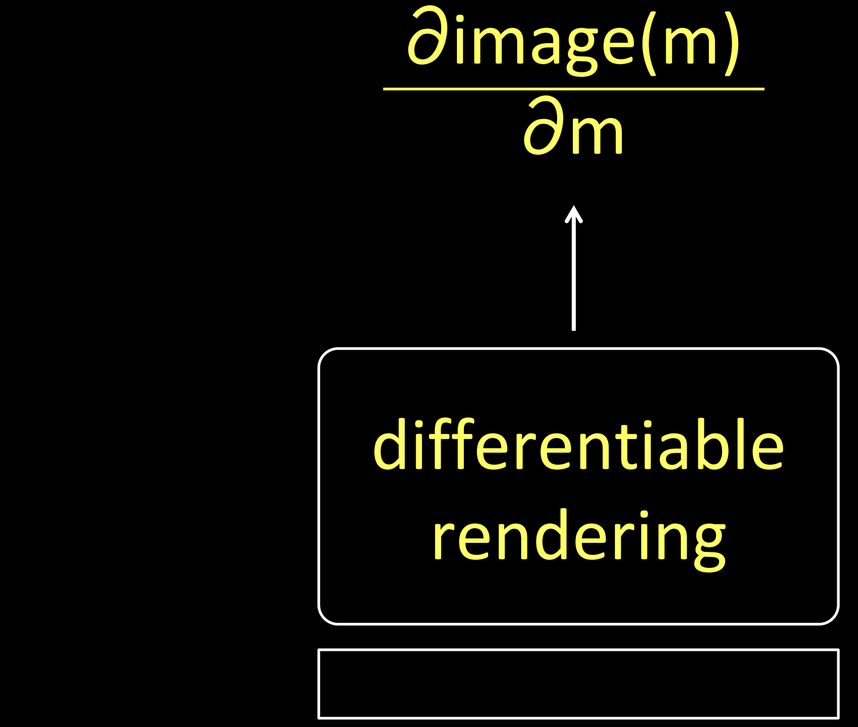
$$\min_{\text{unknowns m (tissue properties)}} \left\| \text{  } - \text{image(m)} \right\|^2$$

solve with gradient descent

while (not converged)

update m with $\frac{\partial \text{loss(m)}}{\partial \text{m}}$

computed with differentiable rendering



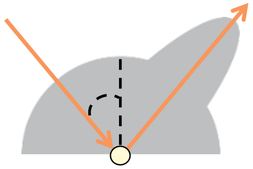
Analysis by synthesis (a.k.a. inverse rendering)

Analysis-by-synthesis optimization:

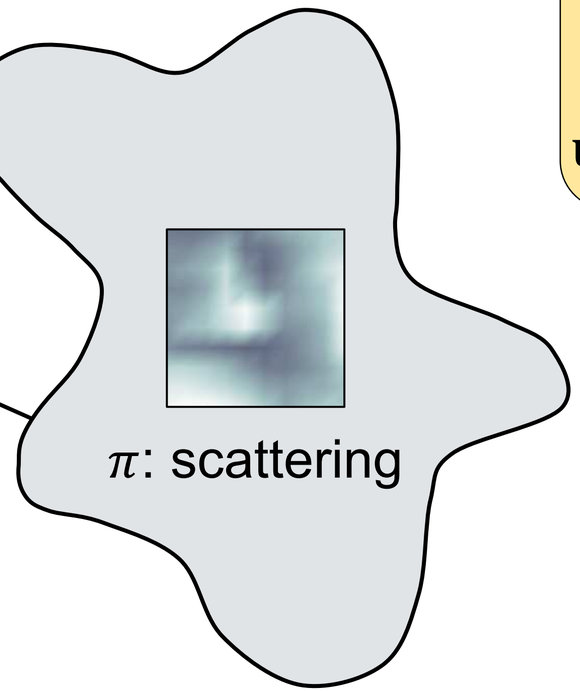
$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{  }\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Analysis by synthesis (a.k.a. inverse rendering)

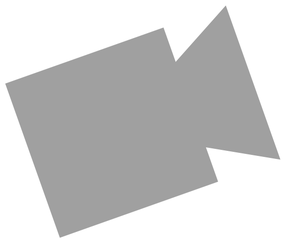
π: illumination



π: BRDF



π: scattering



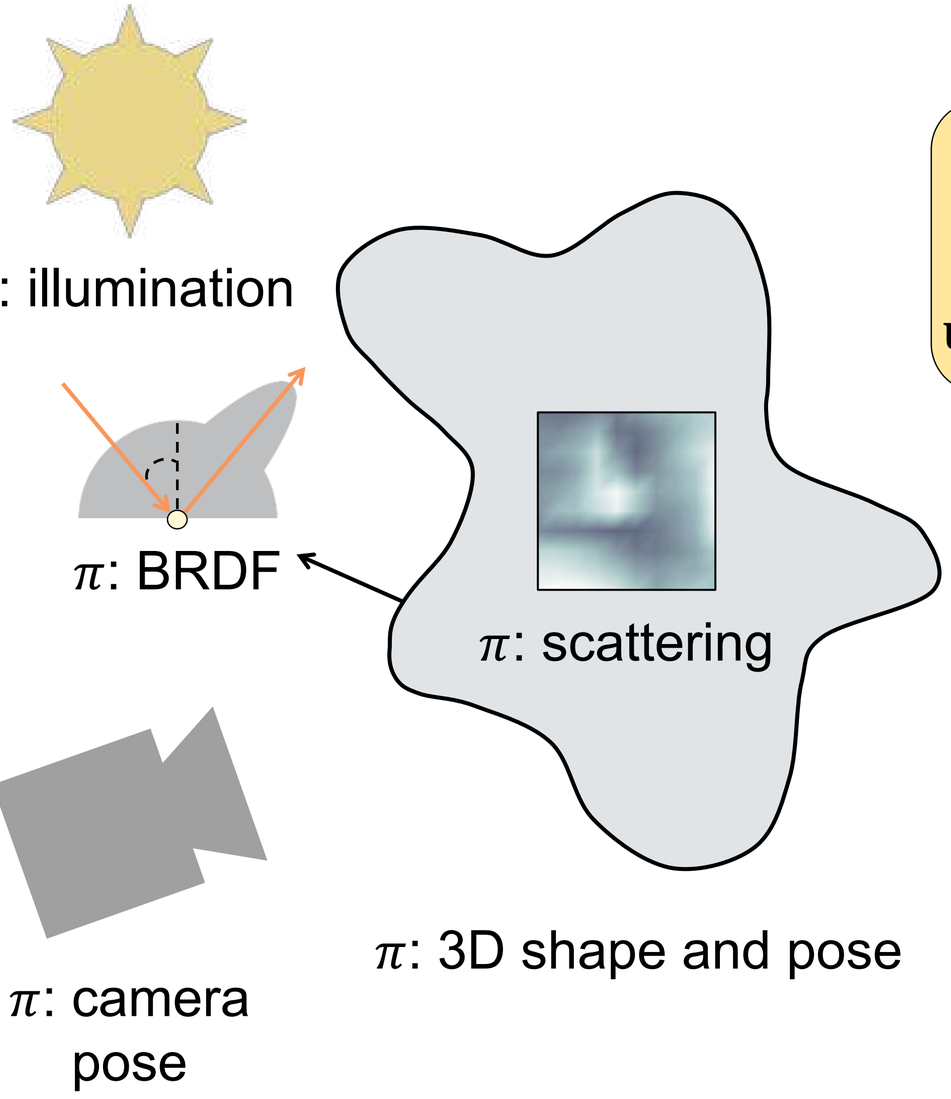
π: camera pose

π: 3D shape and pose

Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{image}\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Analysis by synthesis (a.k.a. inverse rendering)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene}\\ \text{unknowns } \pi}} \text{loss}\left[\ \text{image}\ , \text{render}\left(\begin{matrix}\text{scene}\\ \text{unknowns } \pi\end{matrix}\right)\right]$$

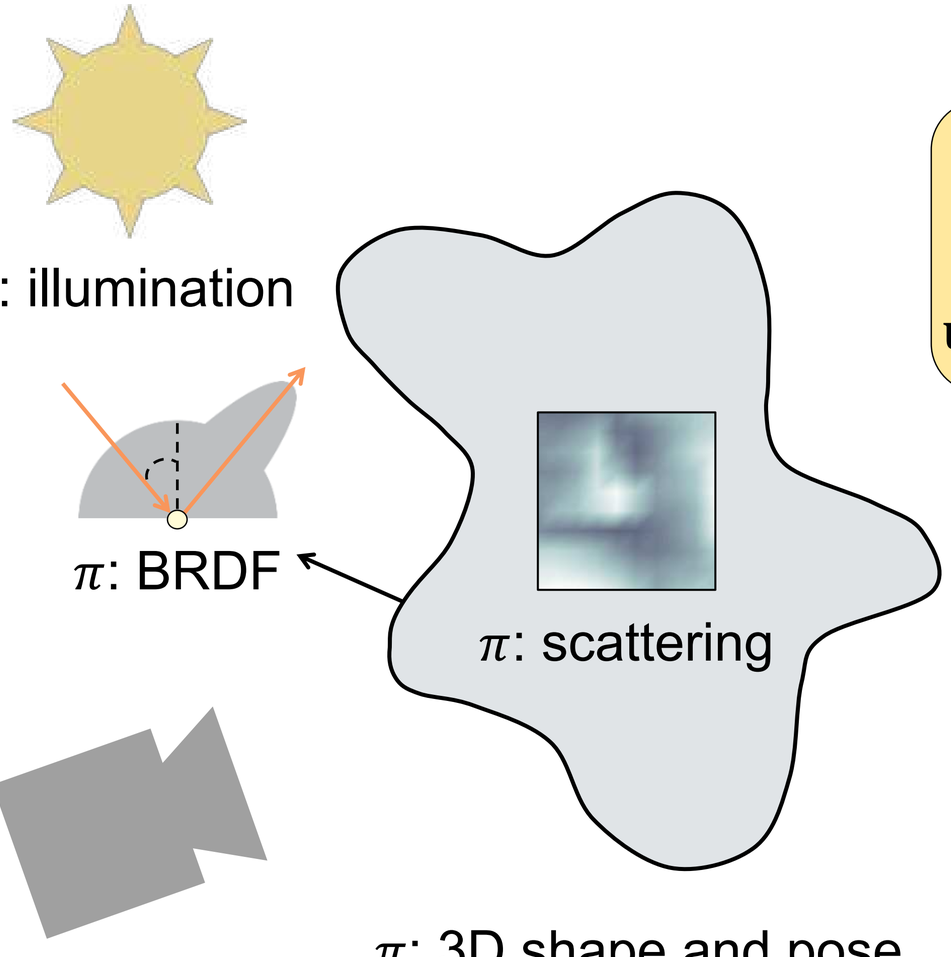
Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \frac{\text{dloss}(\pi)}{\text{d}\pi}$

Analysis by synthesis (a.k.a. inverse rendering)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene}\\ \text{unknowns } \pi}} \text{loss}\left[\ \text{image}\ , \text{render}\left(\begin{matrix}\text{scene}\\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \dfrac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

How do we differentiate light transport?

sensor weight

source weight

$$\text{image} \quad I = \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) T(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}$$

light path

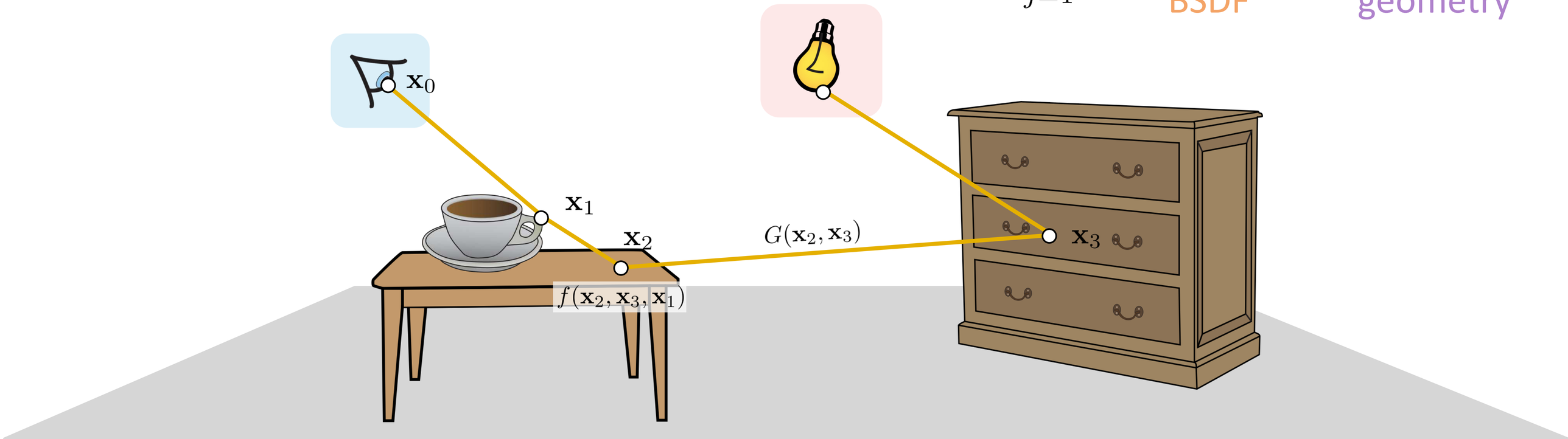
space of all light paths

path throughput

$$T(\bar{\mathbf{x}}) = G(\mathbf{x}_0, \mathbf{x}_1) \prod_{j=1}^{k-1} f(\mathbf{x}_j, \mathbf{x}_{j+1}, \mathbf{x}_{j-1}) G(\mathbf{x}_j, \mathbf{x}_{j+1})$$

BSDF

geometry



REMINDER (?) FROM CALCULUS

Reminder from calculus

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)}^{b(\pi)} f(x,\pi)\mathrm{d}x \overset{?}{=}$$

Reminder from calculus

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)}^{b(\pi)} f(x,\pi)\mathrm{d}x \overset{?}{=}$$

Reminder from calculus

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\mathrm{d}x \overset{?}{=}$$

Reminder from calculus

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\mathrm{d}x$$

$$+ \; f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(\alpha(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

$$+ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

Reminder from calculus

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\mathrm{d}x$$

Move derivative inside integral

$$+ f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(a(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

$$+ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

Reminder from calculus

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\mathrm{d}x$$

Move derivative inside integral

Account for changes in integration limits

$$+ \; f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(\alpha(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

$$+ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

Reminder from calculus

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\mathrm{d}x$$

Move derivative inside integral

Account for changes in integration limits

$$+\ f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(\alpha(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

Account for discontinuities of integrand that depend on π

$$+\ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

A simple example

$$f(x,\pi) = \begin{cases} 0 & \text{if } x < 2\pi \\ 1 & \text{if } x \geq 2\pi \end{cases}$$

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_0^{4\pi} f(x,\pi)\mathrm{d}x \quad =$$

A simple example

$$f(x,\pi) = \begin{cases} 0 & \text{if } x < 2\pi \\ 1 & \text{if } x \geq 2\pi \end{cases}$$

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_0^{4\pi} f(x,\pi)\mathrm{d}x \quad =$$

A simple example

$$f(x,\pi) = \begin{cases} 0 & \text{if } x < 2\pi \\ 1 & \text{if } x \geq 2\pi \end{cases}$$

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_0^{4\pi} f(x,\pi)\mathrm{d}x = \int_0^{2\pi} \frac{\mathrm{d}}{\mathrm{d}\pi} 0\mathrm{d}x + \int_{2\pi}^{4\pi} \frac{\mathrm{d}}{\mathrm{d}\pi} 1\mathrm{d}x$$

Move derivative inside integral

A simple example

$$f(x,\pi) = \begin{cases} 0 & \text{if } x < 2\pi \\ 1 & \text{if } x \geq 2\pi \end{cases}$$

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{0}^{4\pi} f(x,\pi)\mathrm{d}x = \int_{0}^{2\pi}\frac{\mathrm{d}}{\mathrm{d}\pi}0\mathrm{d}x + \int_{2\pi}^{4\pi}\frac{\mathrm{d}}{\mathrm{d}\pi}1\mathrm{d}x$$

Move derivative inside integral

Account for changes in integration limits

$$+\ 1\frac{\mathrm{d}(4\pi)}{\mathrm{d}\pi} - 0\frac{\mathrm{d}0}{\mathrm{d}\pi}$$

A simple example

$$f(x,\pi) = \begin{cases} 0 & \text{if } x < 2\pi \\ 1 & \text{if } x \geq 2\pi \end{cases}$$

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_0^{4\pi} f(x,\pi)\mathrm{d}x = \int_0^{2\pi} \frac{\mathrm{d}}{\mathrm{d}\pi} 0\mathrm{d}x + \int_{2\pi}^{4\pi} \frac{\mathrm{d}}{\mathrm{d}\pi} 1\mathrm{d}x$$

Move derivative inside integral

Account for changes in integration limits

$$+ \; 1\frac{\mathrm{d}(4\pi)}{\mathrm{d}\pi} - 0\frac{\mathrm{d}0}{\mathrm{d}\pi}$$

Account for discontinuities of integrand that depend on π

$$+ \; (0-1)\frac{\mathrm{d}(2\pi)}{\mathrm{d}\pi}$$

Leibniz integral rule

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\,\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\,\mathrm{d}x$$

Move derivative inside integral

Account for changes in integration limits

$$+\ f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(\alpha(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

Account for discontinuities of integrand that depend on π

$$+\ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

Leibniz integral rule

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\,\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\,\mathrm{d}x$$

Move derivative inside integral

Account for changes in integration limits

$$+\ f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(\alpha(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

Account for discontinuities of integrand that depend on π

$$+\ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

Leibniz integral rule

Differentiation under the integral sign
Also known as the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{a(\pi)} f(x,\pi)\,\mathrm{d}x = \int_{a(\pi)}^{b(\pi)} \frac{\mathrm{d}}{\mathrm{d}\pi} f(x,\pi)\,\mathrm{d}x$$

Interior integral

Move derivative inside integral

Boundary terms

Account for changes in integration limits

$$+\ f(b(\pi),\pi)\frac{\mathrm{d}b(\pi)}{\mathrm{d}\pi} - f(\alpha(\pi);\pi)\frac{\mathrm{d}a(\pi)}{\mathrm{d}\pi}$$

Account for discontinuities of integrand that depend on π

$$+\ \sum_i \left(f(c_i(\pi)^-,\pi) - f(c_i(\pi)^+,\pi)\right)\frac{\mathrm{d}c_i(\pi)}{\mathrm{d}\pi}$$

Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) \overset{?}{=}$$

Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) \overset{?}{=}$$

Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) \overset{?}{=}$$

Reynolds transport theorem [1903]
Generalization of the Leibniz rule

Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) = \int_{\Omega(\pi)} \frac{\mathrm{d}f(x,\pi)}{\mathrm{d}\pi}\,\mathrm{d}A(x) + \int_{\partial\Omega(\pi)} g(x,\pi)\,\mathrm{d}l(x)$$

Reynolds transport theorem [1903]
Generalization of the Leibniz rule

Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) = \int_{\Omega(\pi)} \frac{\mathrm{d}f(x,\pi)}{\mathrm{d}\pi}\,\mathrm{d}A(x) + \int_{\partial\Omega(\pi)} g(x,\pi)\,\mathrm{d}l(x)$$

Reynolds transport theorem [1903]
Generalization of the Leibniz rule

Boundary domain

||

discontinuity points ∪ boundary of domain Ω
(if they depend on π)

Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) = \int_{\Omega(\pi)} \frac{\mathrm{d}f(x,\pi)}{\mathrm{d}\pi}\,\mathrm{d}A(x) + \int_{\partial\Omega(\pi)} g(x,\pi)\,\mathrm{d}l(x)$$

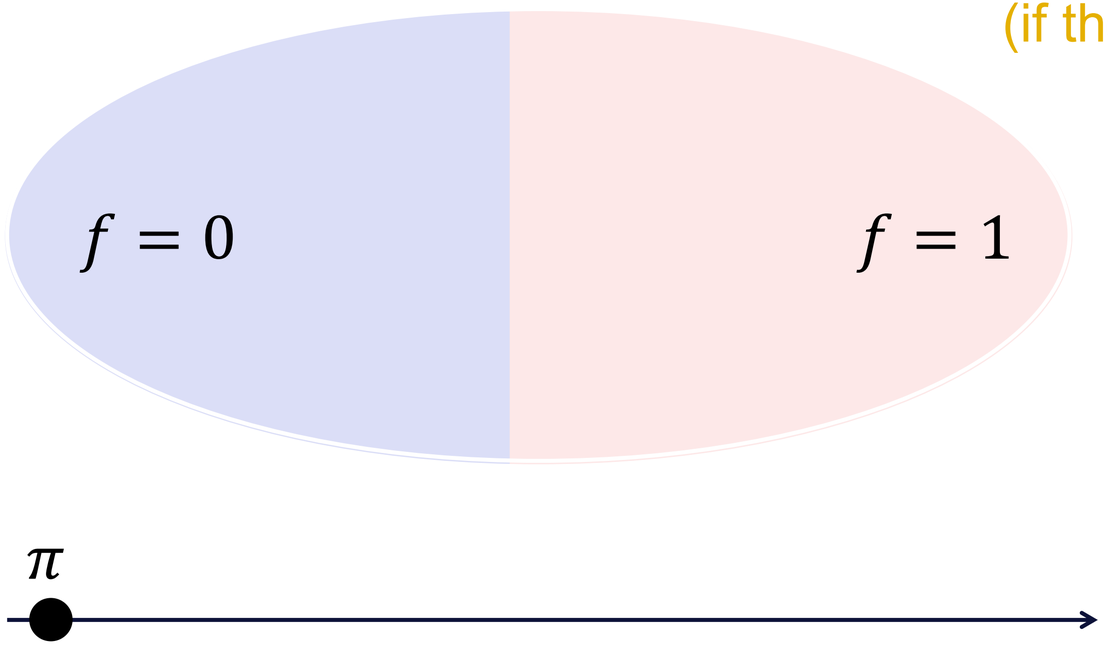
Boundary domain

||

discontinuity points ∪ boundary of domain Ω

(if they depend on π)

Reynolds transport theorem [1903]
Generalization of the Leibniz rule



Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) = \int_{\Omega(\pi)} \frac{\mathrm{d}f(x,\pi)}{\mathrm{d}\pi}\,\mathrm{d}A(x) + \int_{\partial\Omega(\pi)} g(x,\pi)\,\mathrm{d}l(x)$$

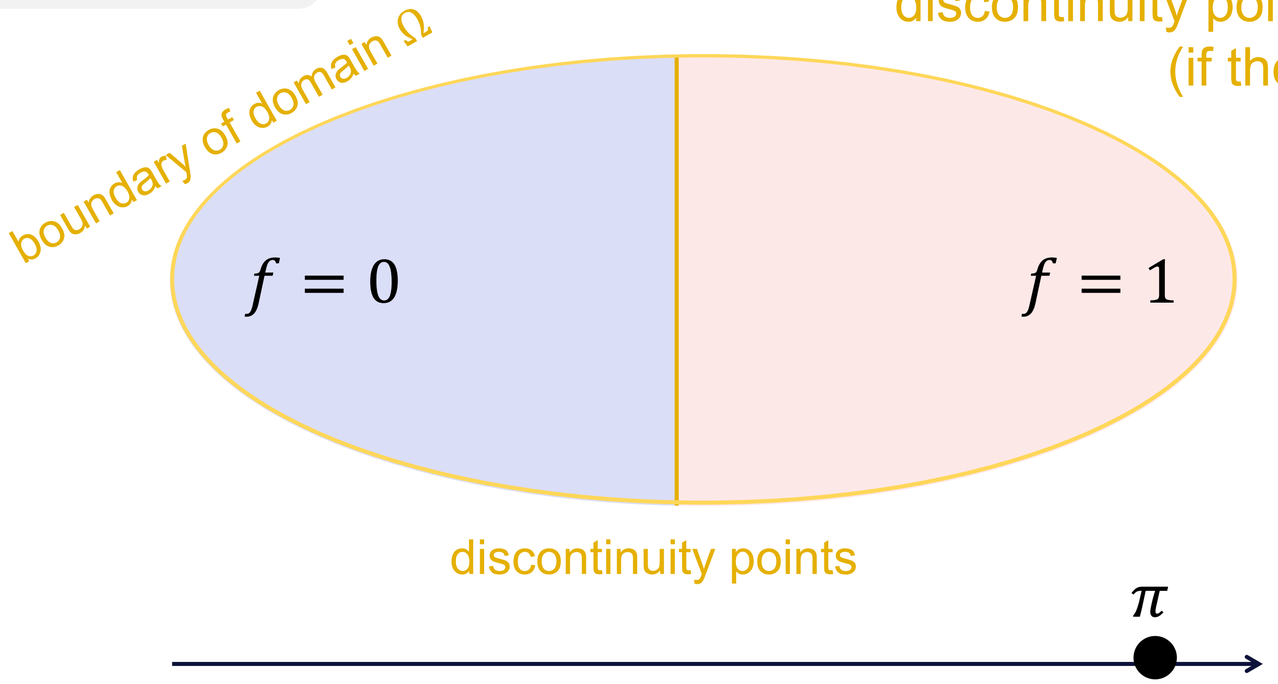
Boundary domain

||

discontinuity points ∪ boundary of domain Ω

(if they depend on π)

Reynolds transport theorem [1903]
Generalization of the Leibniz rule



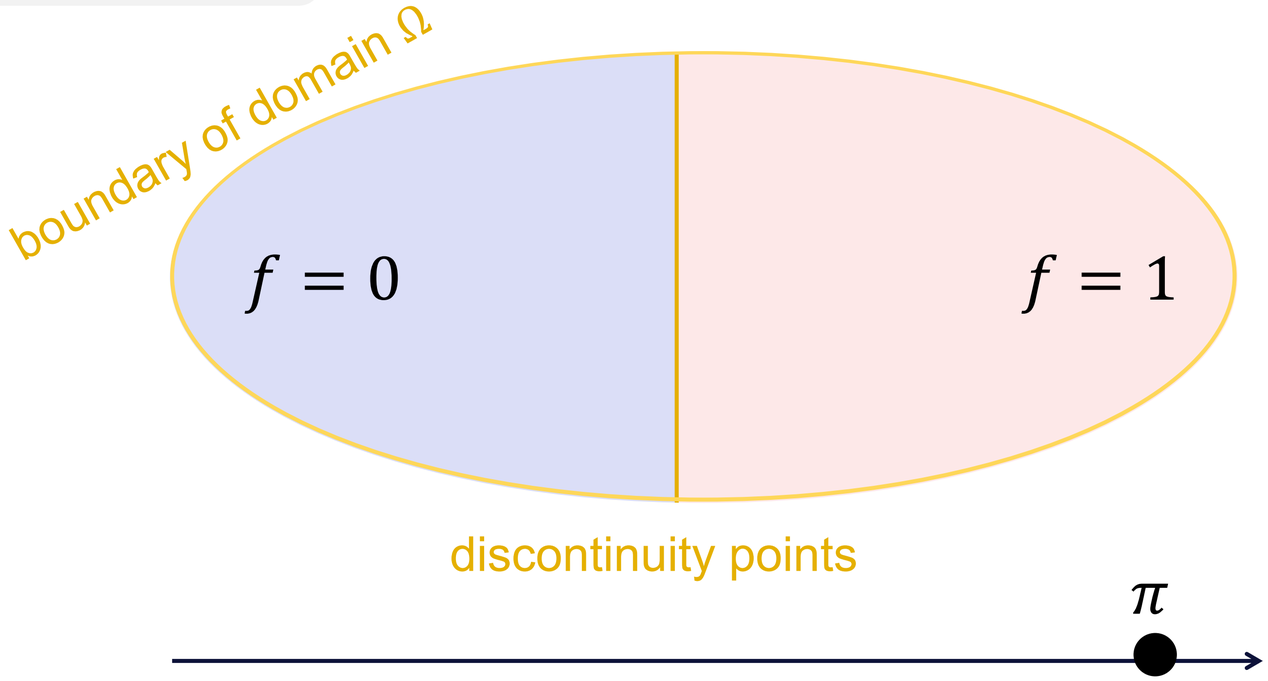
Reynolds transport theorem

$$\frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\Omega(\pi)} f(x,\pi)\,\mathrm{d}A(x) = \int_{\Omega(\pi)} \frac{\mathrm{d}f(x,\pi)}{\mathrm{d}\pi}\,\mathrm{d}A(x) + \int_{\partial\Omega(\pi)} g(x,\pi)\,\mathrm{d}l(x)$$

Reynolds transport theorem [1903]
Generalization of the Leibniz rule

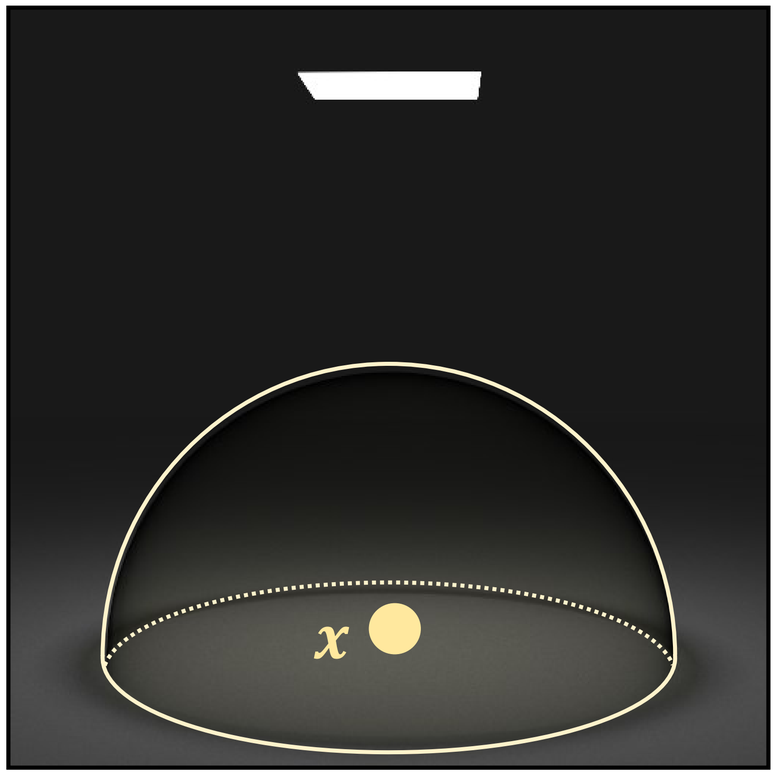
Interior integral

Boundary integral



DIFFERENTIATING DIRECT ILLUMINATION

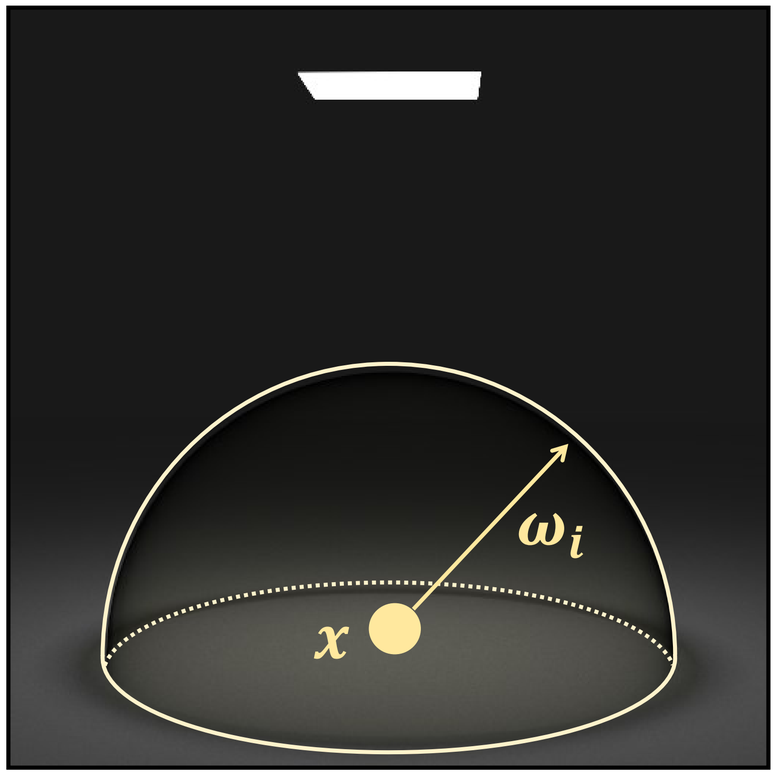
Direct illumination integral



Radiance from x:

$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Direct illumination integral

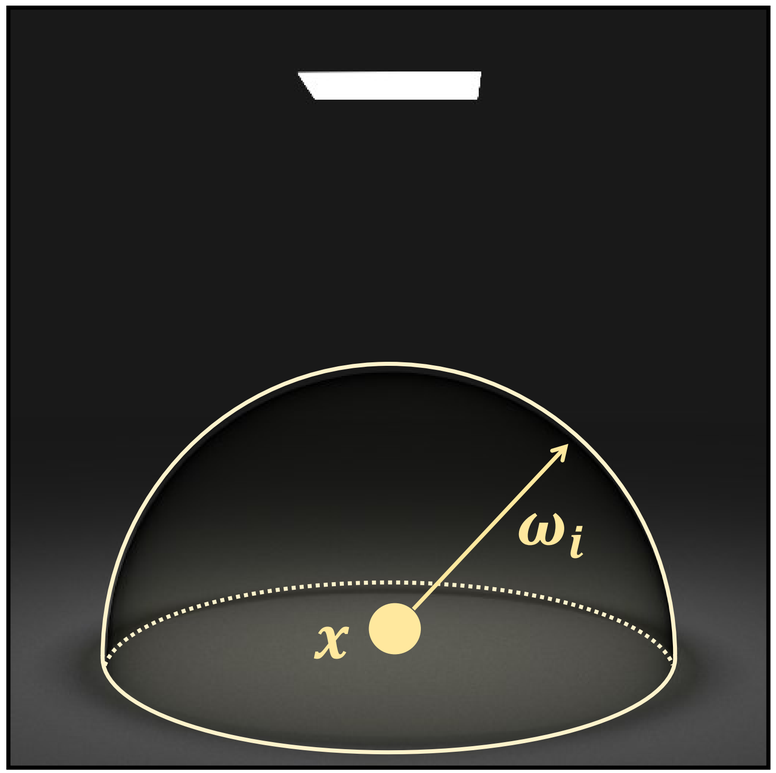


Radiance from x:

$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Unit hemisphere

Direct illumination integral



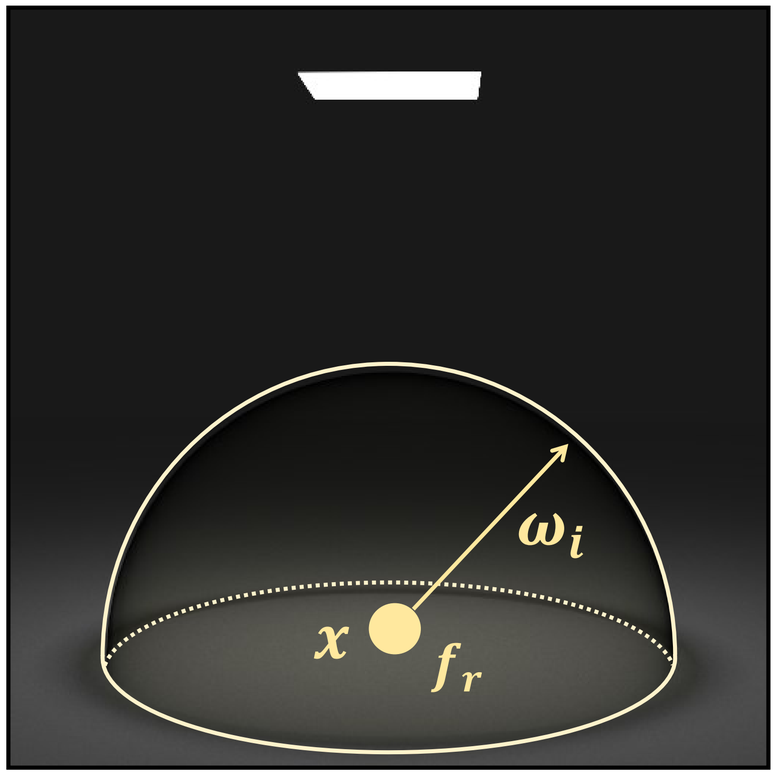
Radiance from x:

$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Incident radiance

Unit hemisphere

Direct illumination integral



Radiance from x:

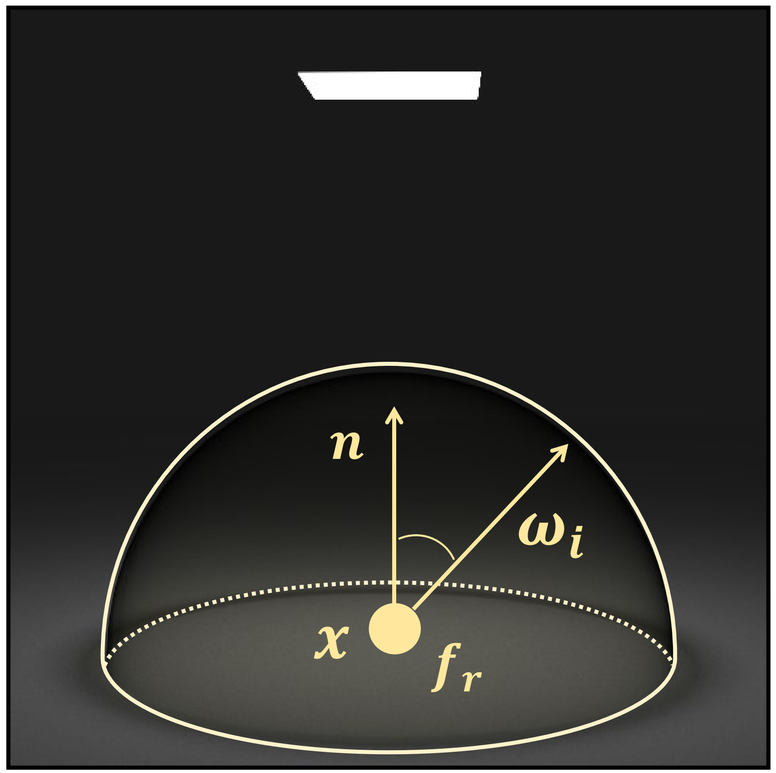
$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Reflectance (BRDF)

Incident radiance

Unit hemisphere

Direct illumination integral



Radiance from x:

$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

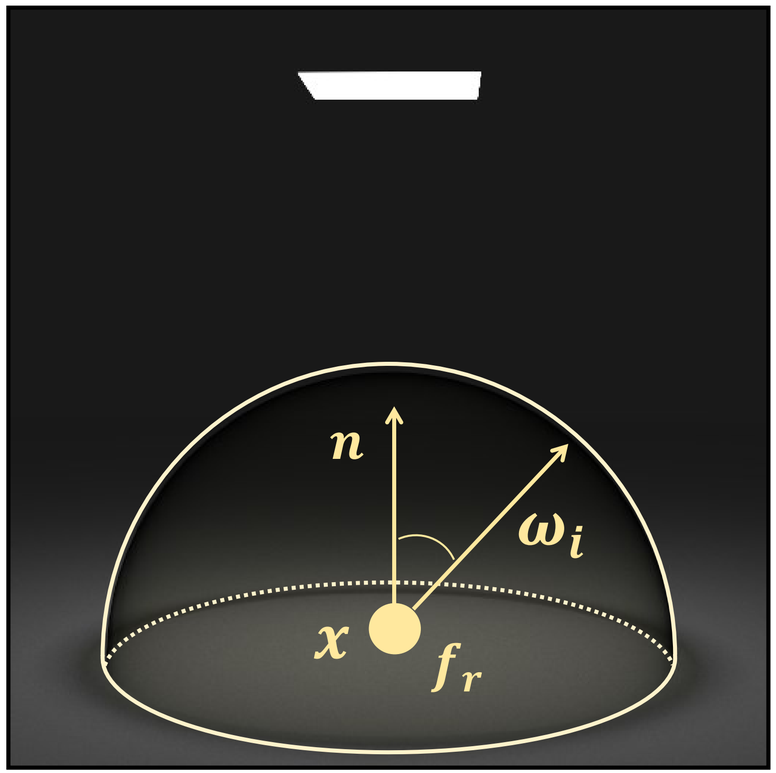
Reflectance (BRDF)

Incident radiance

Shading wrt normal $\boldsymbol{n}$

Unit hemisphere

Direct illumination integral



Radiance from x:

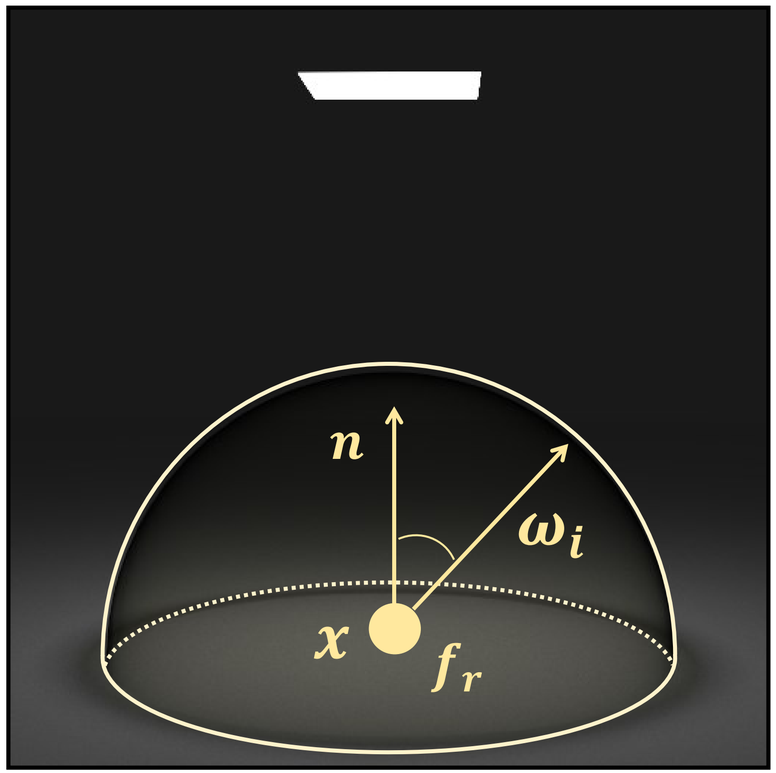
$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Reflectance (BRDF) | Incident radiance | Shading wrt normal $\boldsymbol{n}$

Unit hemisphere

Monte Carlo rendering:

Direct illumination integral



Radiance from x:

$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

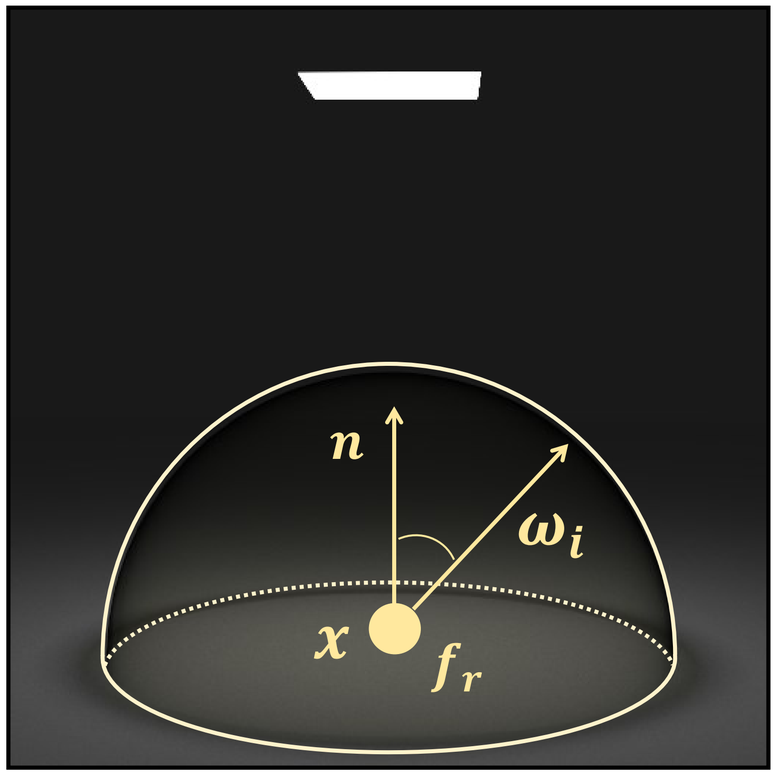
Reflectance (BRDF) | Incident radiance | Shading wrt normal $\boldsymbol{n}$

Unit hemisphere

Monte Carlo rendering:

- Sample random directions ω_i^s from PDF $p(\omega_i)$

Direct illumination integral



Radiance from x:

$$I = \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Reflectance (BRDF) | Incident radiance | Shading wrt normal $\boldsymbol{n}$

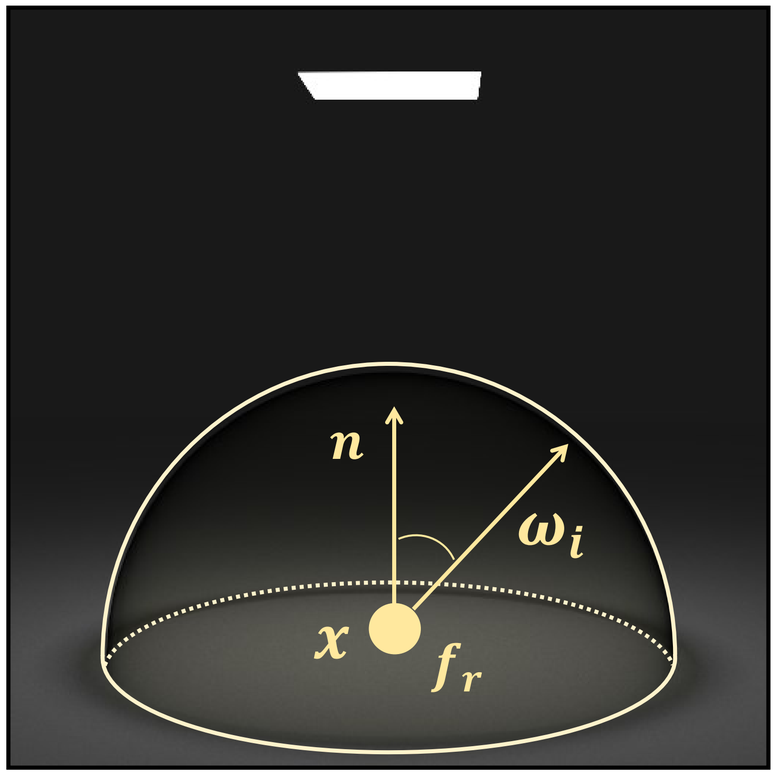
Unit hemisphere

Monte Carlo rendering:

- Sample random directions ω_i^s from PDF $p(\omega_i)$
- Form estimator

$$I \approx \sum_s \frac{f_r(\omega_i^s, \omega_o)\, L_i(\omega_i^s)\, (n \cdot \omega_i^s)}{p(\omega_i^s)}$$

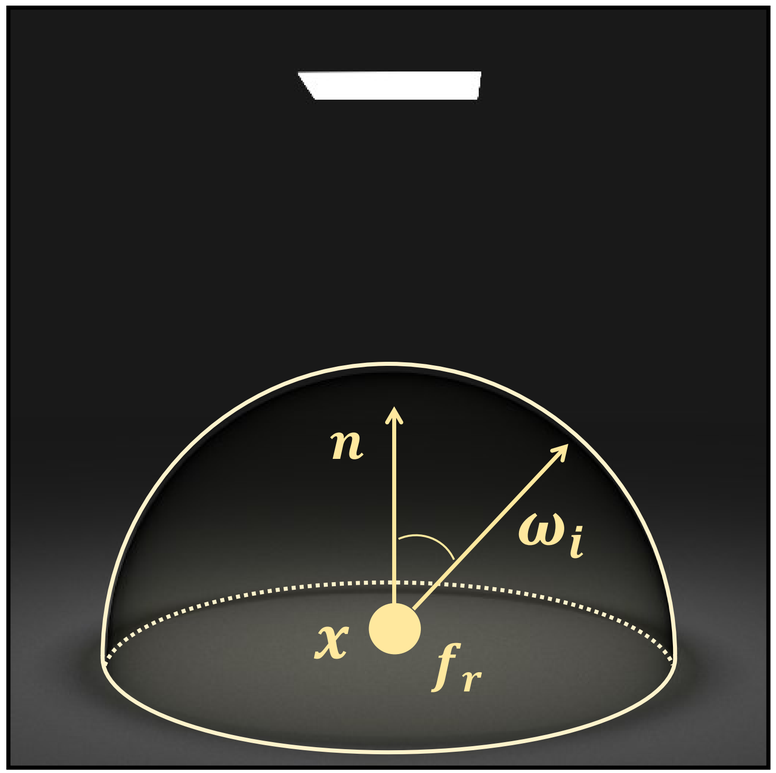
Differential direct illumination



Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \frac{\mathrm{d}}{\mathrm{d}\pi} \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

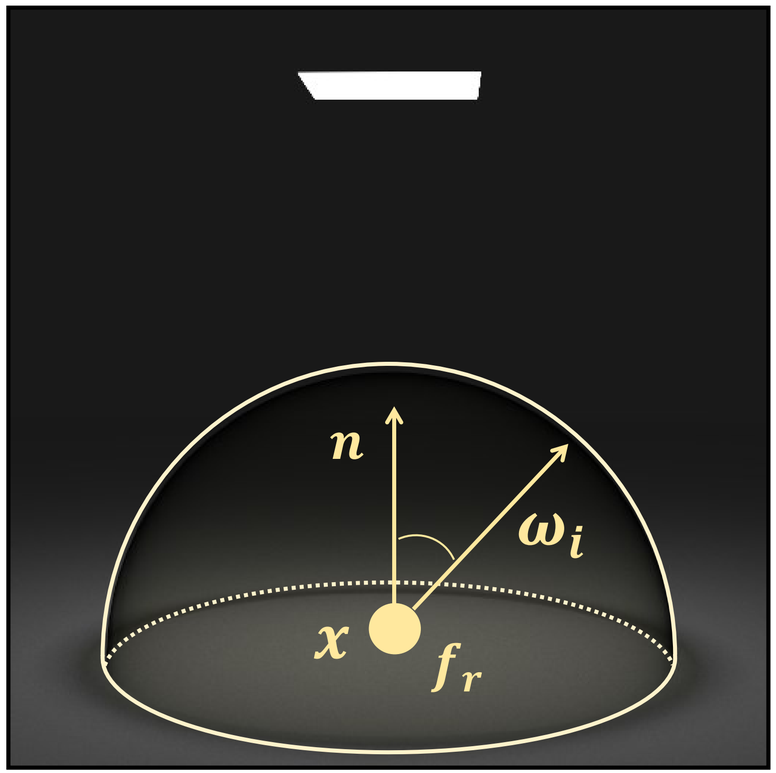
Differential direct illumination: local parameters



Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

Differential direct illumination: local parameters



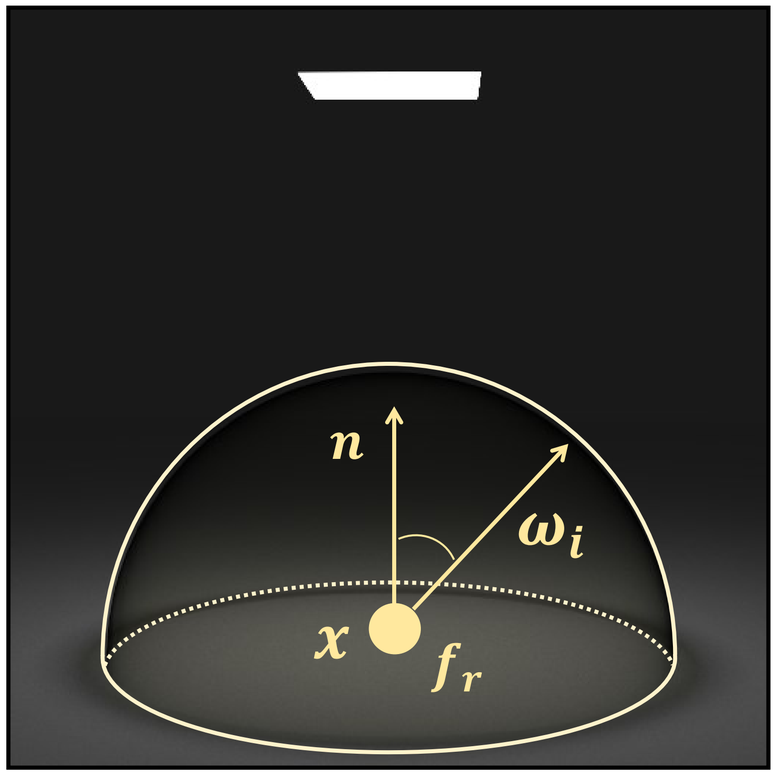
Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \frac{\mathrm{d}}{\mathrm{d}\pi} \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

π: *local* parameters

- BRDF parameters
- *shading* normal
- illumination brightness

Differential direct illumination: local parameters



Differential radiance from x:

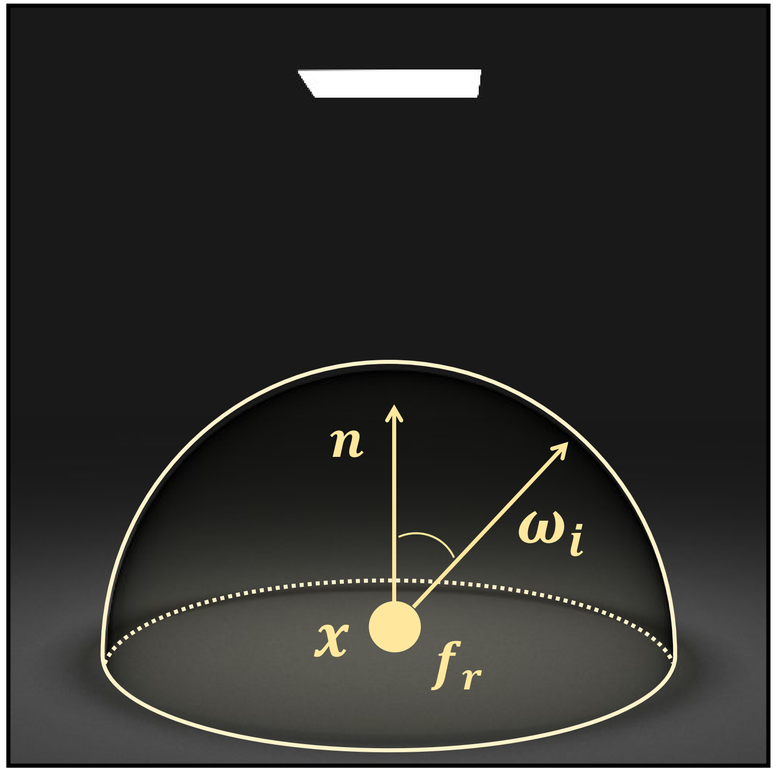
$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathbb{H}^2} \frac{\mathrm{d}}{\mathrm{d}\pi}\{f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\}\, \mathrm{d}\sigma(\omega_i)$$

Just move derivative inside integral

π: *local* parameters

- BRDF parameters
- *shading* normal
- illumination brightness

Differential direct illumination: local parameters



$\boldsymbol{\pi}$: *local* parameters
- BRDF parameters
- *shading* normal
- illumination brightness

Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathbb{H}^2} \frac{\mathrm{d}}{\mathrm{d}\pi}\{f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\}\, \mathrm{d}\sigma(\omega_i)$$

Just move derivative inside integral

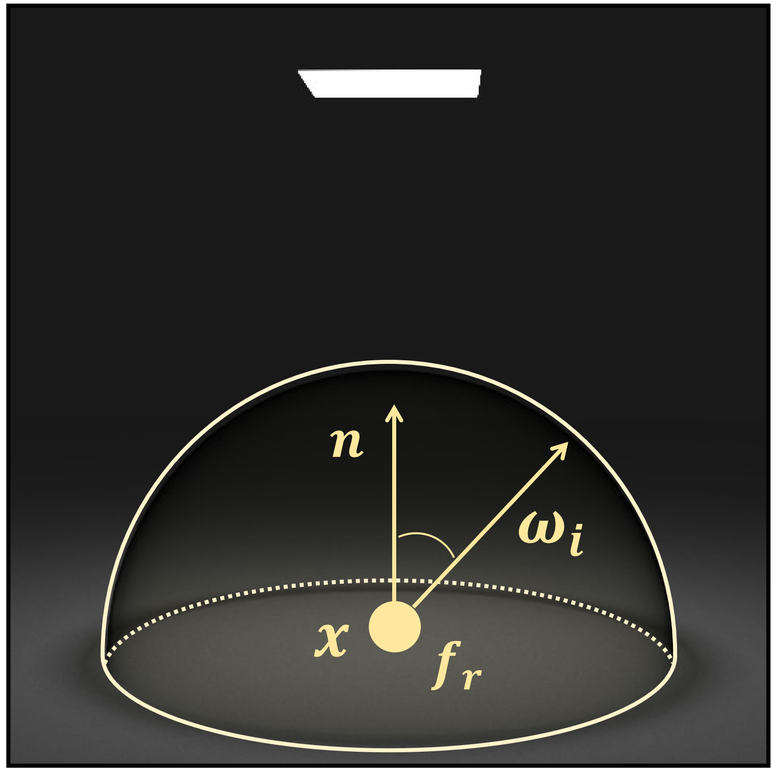
Monte Carlo differentiable rendering:

- Sample random directions ω_i^s from PDF $p(\omega_i)$
- Form estimator

Just differentiate numerator
[Khungurn et al. 2015, Gkioulekas et al. 2015]

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} \approx \sum_s \frac{\frac{\mathrm{d}}{\mathrm{d}\pi}\{f_r(\omega_i^s, \omega_o)\, L_i(\omega_i^s)\, (n \cdot \omega_i^s)\}}{p(\omega_i^s)}$$

Alternative estimator



π: *local* parameters
- BRDF parameters

Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathbb{H}^2} \frac{\mathrm{d}}{\mathrm{d}\pi} \{f_r(\omega_i, \omega_o, \pi) L_i(\omega_i)(n \cdot \omega_i)\} \mathrm{d}\sigma(\omega_i)$$

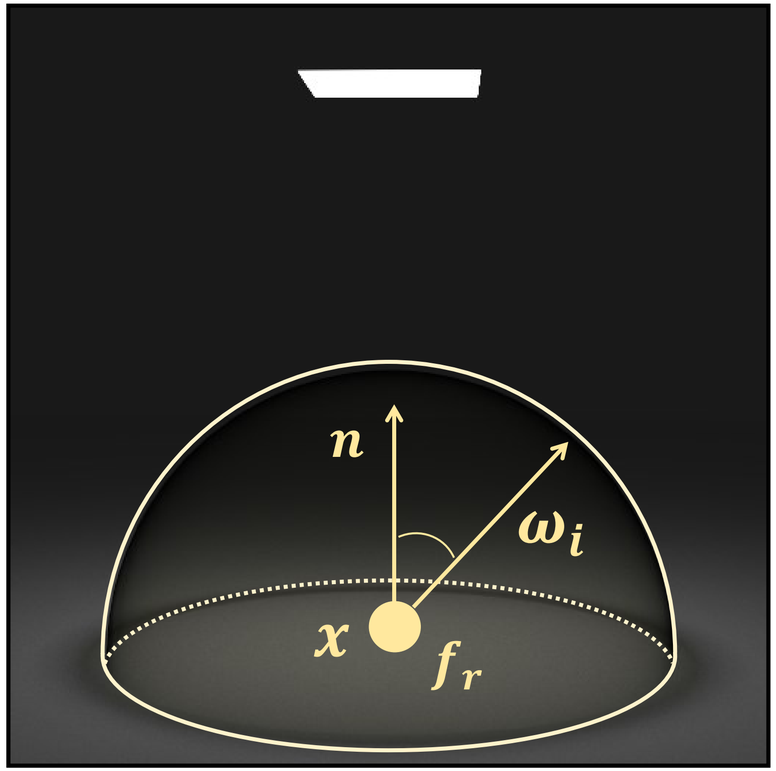
Just move derivative inside integral

Monte Carlo estimation:

- Sample random directions ω_i^s from PDF $p(\omega_i, \pi)$
- Form estimator

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} \approx \sum_s \frac{\frac{\mathrm{d}}{\mathrm{d}\pi} \{f_r(\omega_i^s, \omega_o, \pi)\ L_i(\omega_i^s)\ (n \cdot \omega_i^s)\}}{p(\omega_i^s, \pi)}$$

Alternative estimator



π: *local* parameters
- BRDF parameters

Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathbb{H}^2} \frac{\mathrm{d}}{\mathrm{d}\pi} \{f_r(\omega_i, \omega_o, \pi) L_i(\omega_i)(n \cdot \omega_i)\} \mathrm{d}\sigma(\omega_i)$$

Just move derivative inside integral

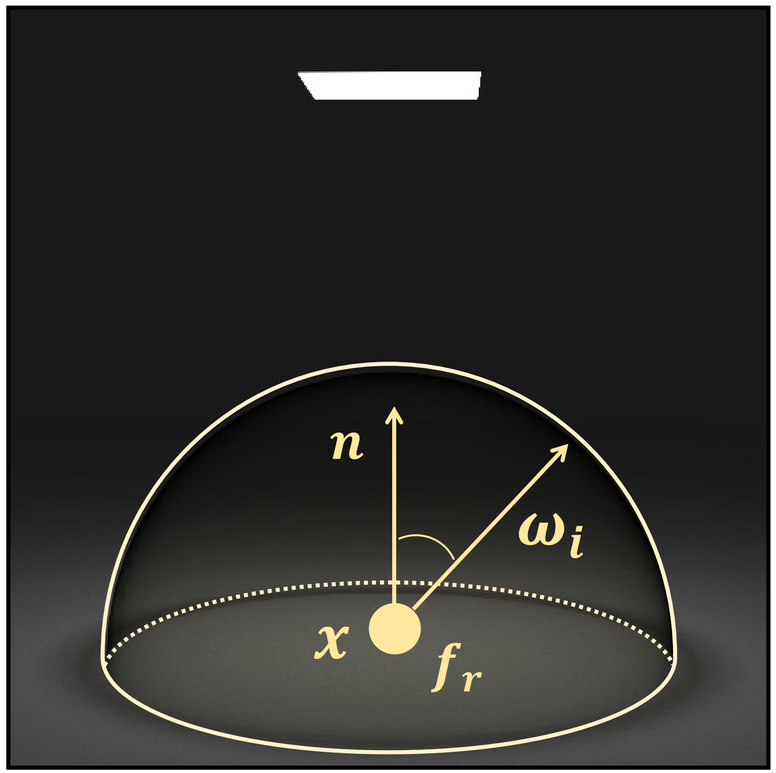
Monte Carlo estimation:

- Sample random directions ω_i^s from PDF $p(\omega_i, \pi)$
- Form estimator

Differentiate entire contribution
[Zeltner et al. 2021]

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} \approx \sum_s \frac{\mathrm{d}}{\mathrm{d}\pi} \left\{ \frac{f_r(\omega_i^s, \omega_o, \pi)\, L_i(\omega_i^s)\, (n \cdot \omega_i^s)}{p(\omega_i^s, \pi)} \right\}$$

Differential direct illumination: global parameters



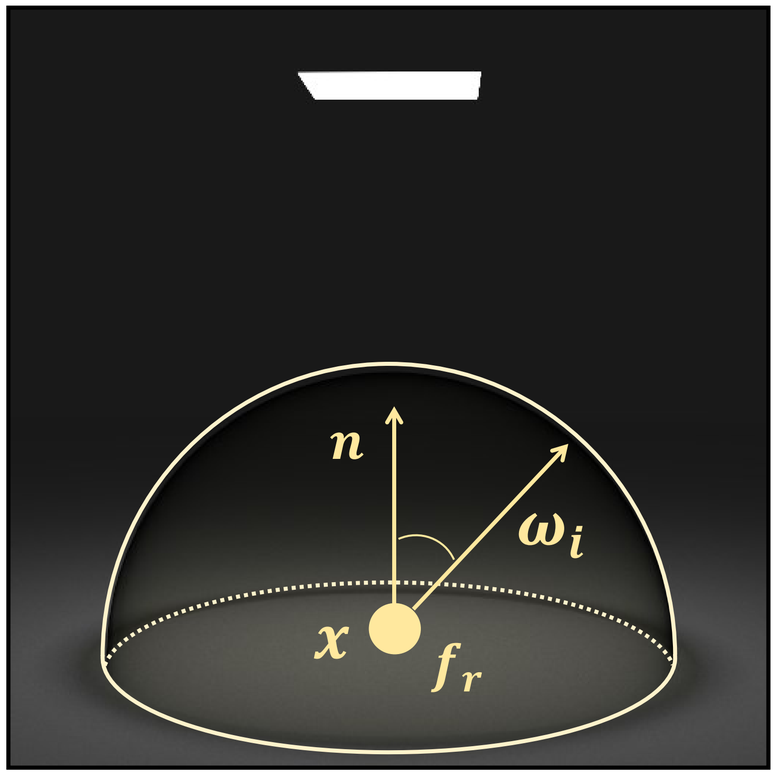
Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \frac{\mathrm{d}}{\mathrm{d}\pi} \int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

π: *global* parameters

- shape and pose of different scene elements (camera, sources, objects)

Differential direct illumination: global parameters



Differential radiance from x:

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \frac{\mathrm{d}}{\mathrm{d}\pi}\int_{\mathbb{H}^2} f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\, \mathrm{d}\sigma(\omega_i)$$

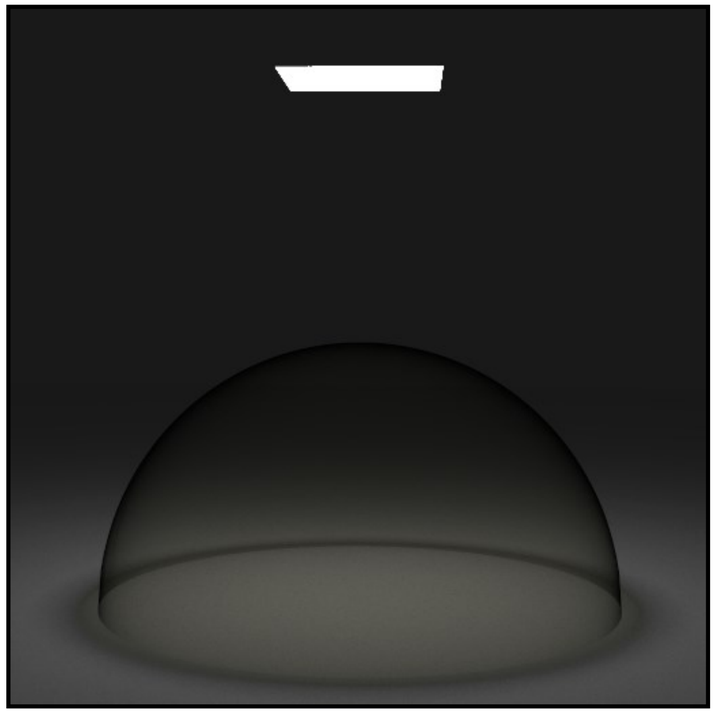
$$= \int_{\mathbb{H}^2} \frac{\mathrm{d}}{\mathrm{d}\pi}\{f_r(\omega_i, \omega_o)\, L_i(\omega_i)\, (n \cdot \omega_i)\}\, \mathrm{d}\sigma(\omega_i)$$

Need to use full Reynolds transport theorem

π: *global* parameters

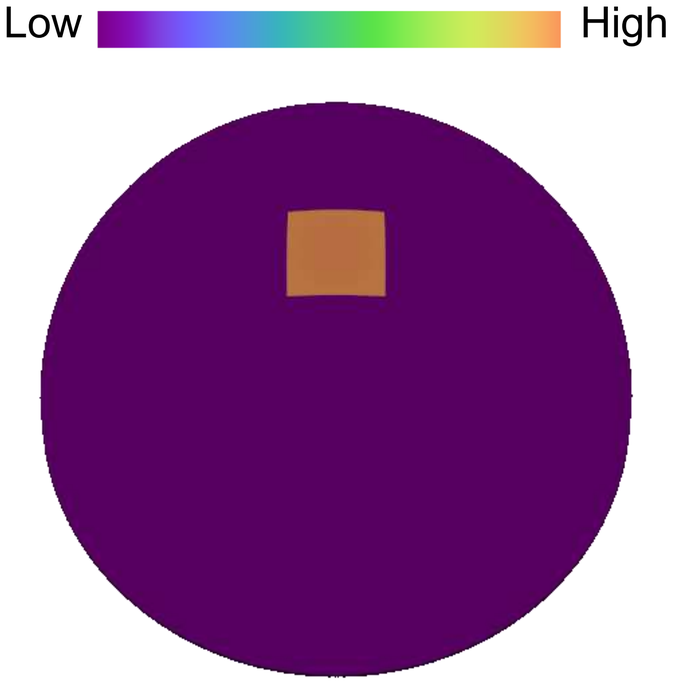
- shape and pose of different scene elements (camera, sources, objects)

Discontinuities in the integrand

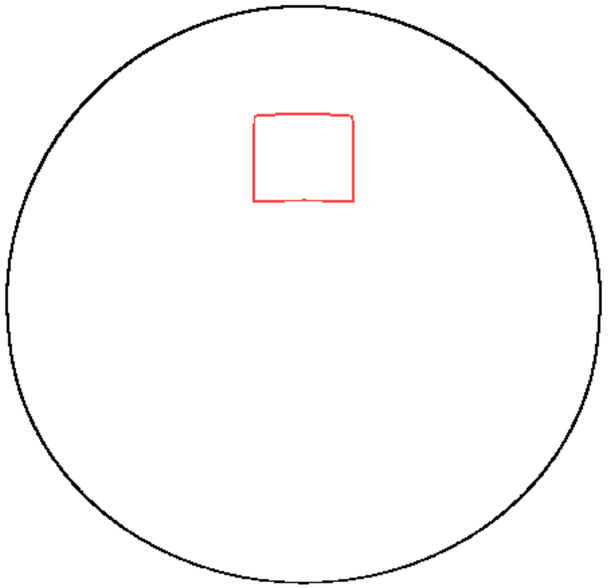


π: size of the emitter

$$I = \int_{\mathbb{H}^2} \underbrace{f_r(\omega_i, \omega_o) L_i(\omega_i)(n \cdot \omega_i)}_{f(\omega_i)} \mathrm{d}\sigma(\omega_i)$$



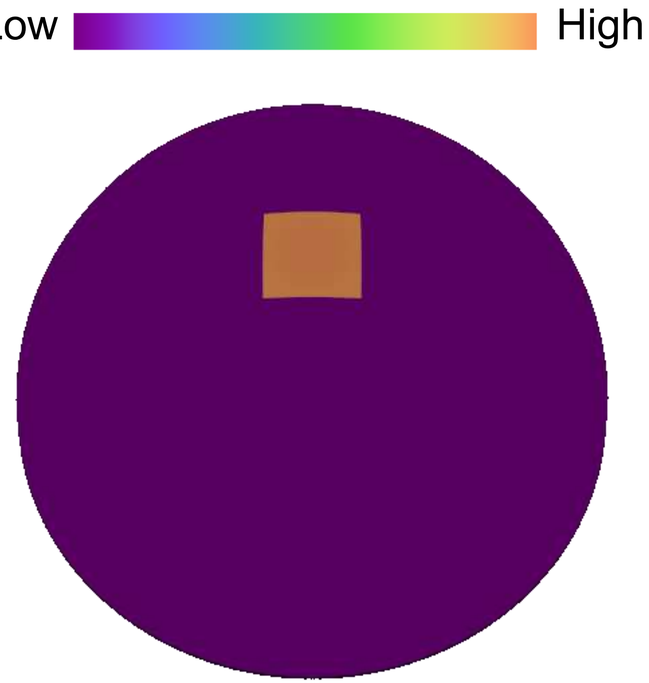
Integrand
$f(\omega_i)$



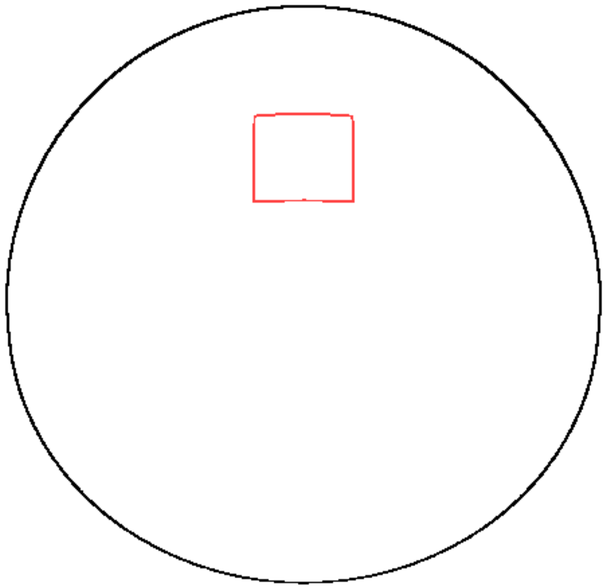
Discontinuous points
(π-dependent)

Applying the Reynolds transport theorem

$$I = \int_{\mathbb{H}^2} f(\omega_i, \omega_o) \mathrm{d}\sigma(\omega_i)$$



Integrand
$f(\omega_i)$



Discontinuous points
(π-dependent)

Applying the Reynolds transport theorem

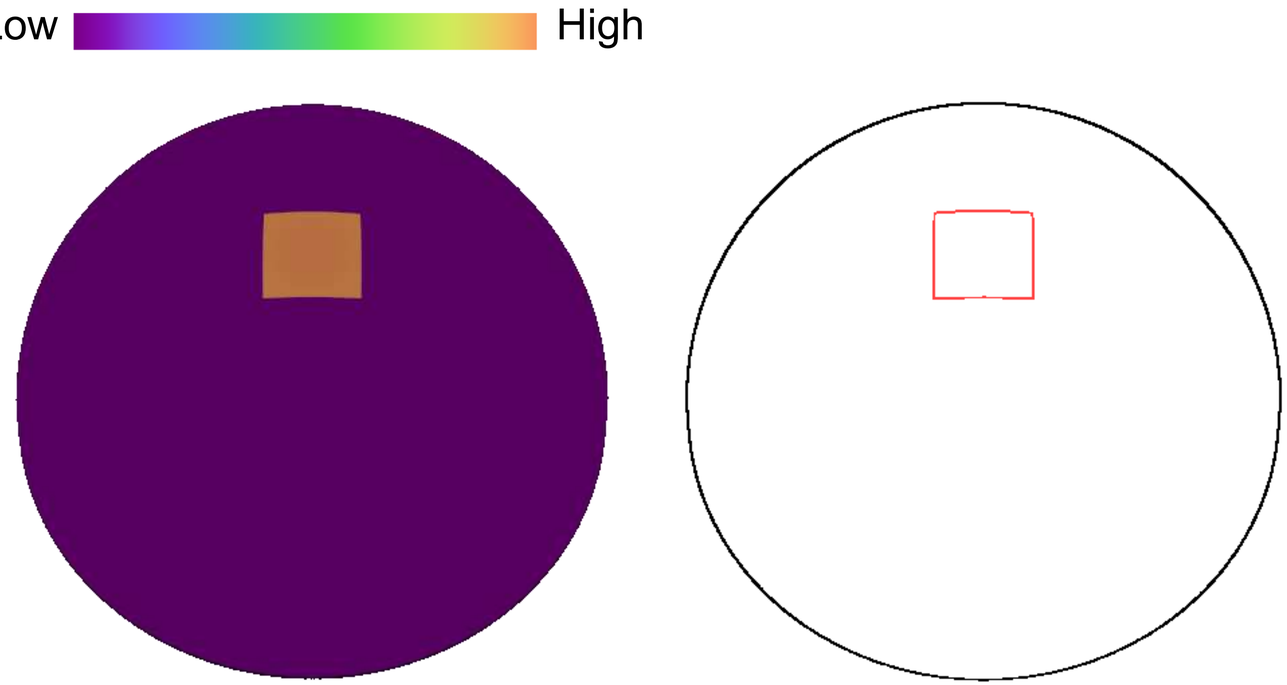
$$I = \int_{\mathbb{H}^2} f(\omega_i, \omega_o) \mathrm{d}\sigma(\omega_i)$$

$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathbb{H}^2} \frac{\mathrm{d}f}{\mathrm{d}\pi} \, \mathrm{d}\sigma + \int_{\partial\mathbb{H}^2} g \, \mathrm{d}l$$

Interior integral (same as for local parameters)

Boundary integral

[Ramamoorthi et al. 2007, Li et al. 2019]

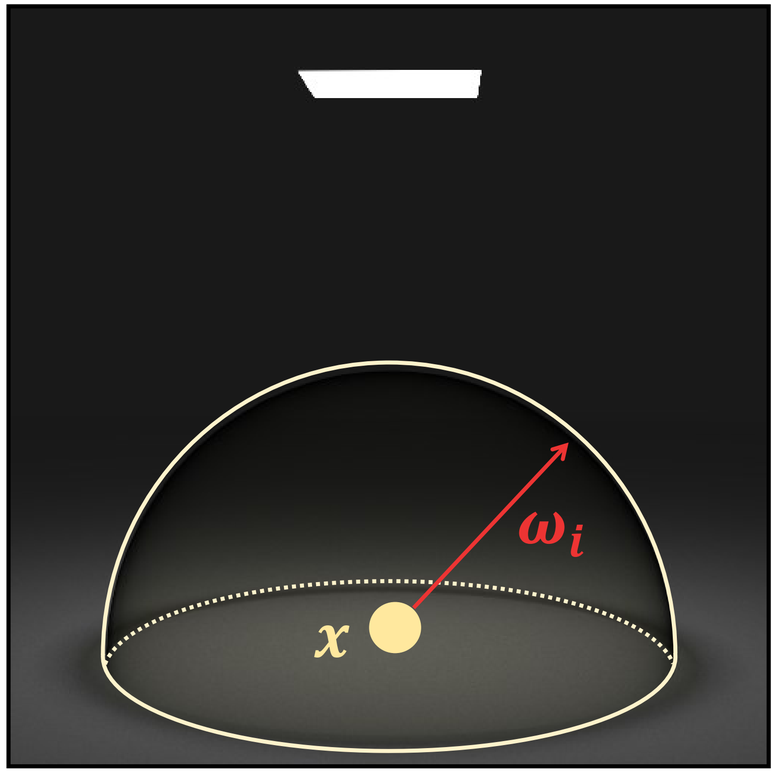


Integrand $f(\omega_i)$

Discontinuous points (π-dependent)

Reparameterizing the direct illumination integral

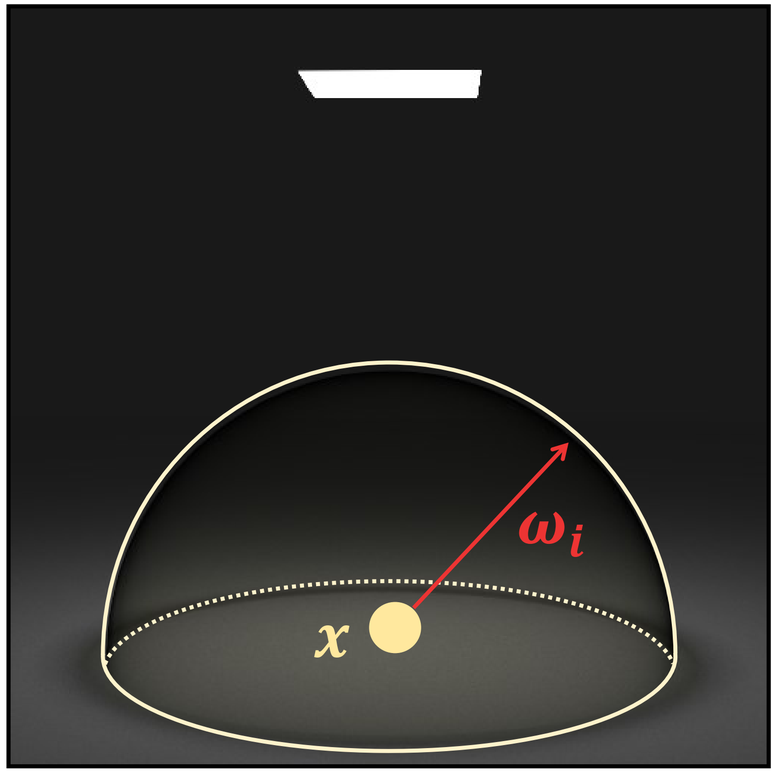
Hemispherical integral



$$I = \int_{\mathbb{H}^2} f(\omega_i)\, \mathrm{d}\sigma(\omega_i)$$

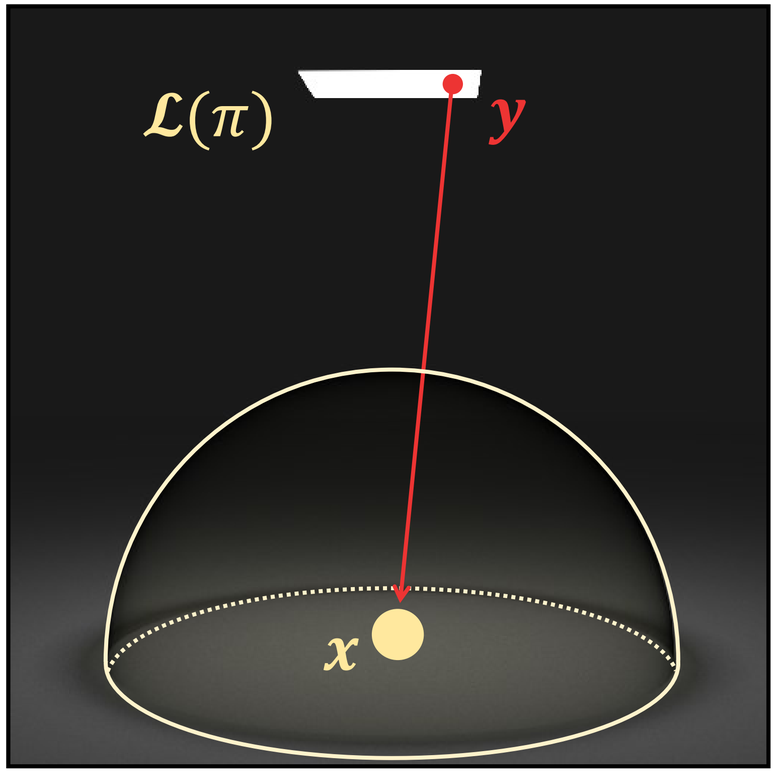
Reparameterizing the direct illumination integral

Hemispherical integral



Change of variables

Surface integral

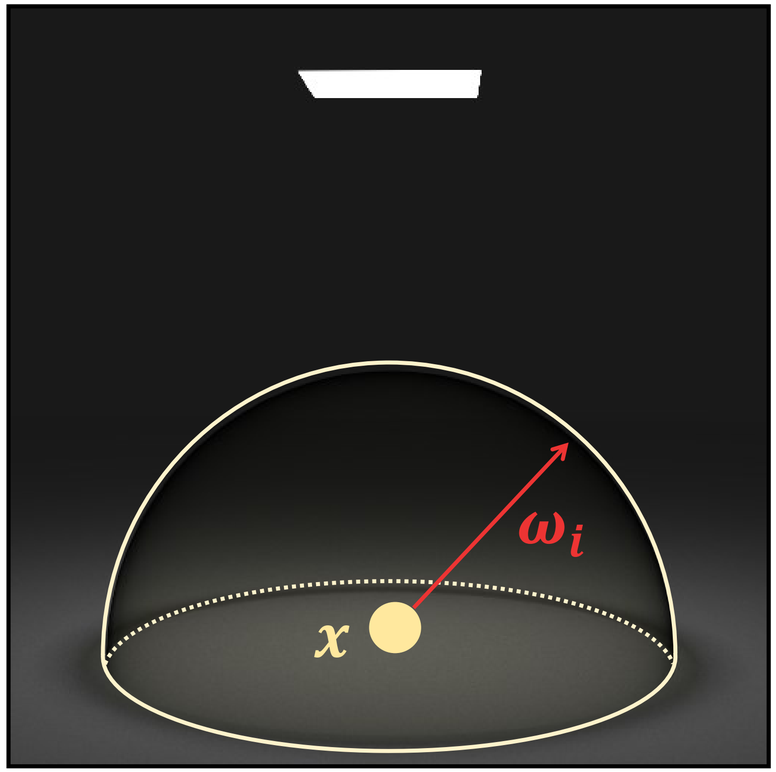


$$I = \int_{\mathbb{H}^2} f(\boldsymbol{\omega_i})\, \mathrm{d}\sigma(\boldsymbol{\omega_i})$$

$$I = \int_{\mathcal{L}(\pi)} f(\boldsymbol{y} \to \boldsymbol{x})\, G(\boldsymbol{x}, \boldsymbol{y})\, \mathrm{d}A(\boldsymbol{y})$$

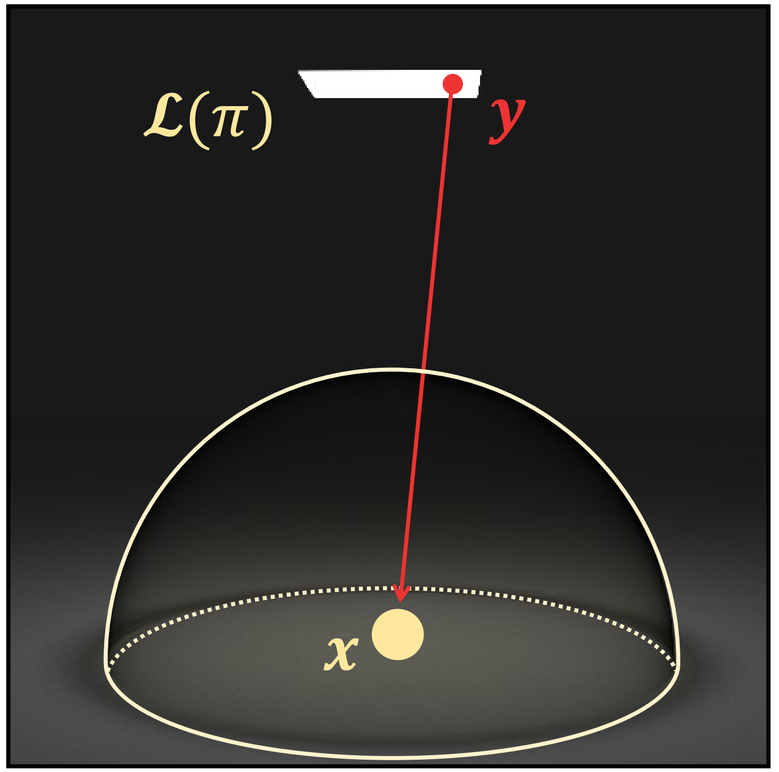
Reparameterizing the direct illumination integral

Hemispherical integral



Change of variables

Surface integral



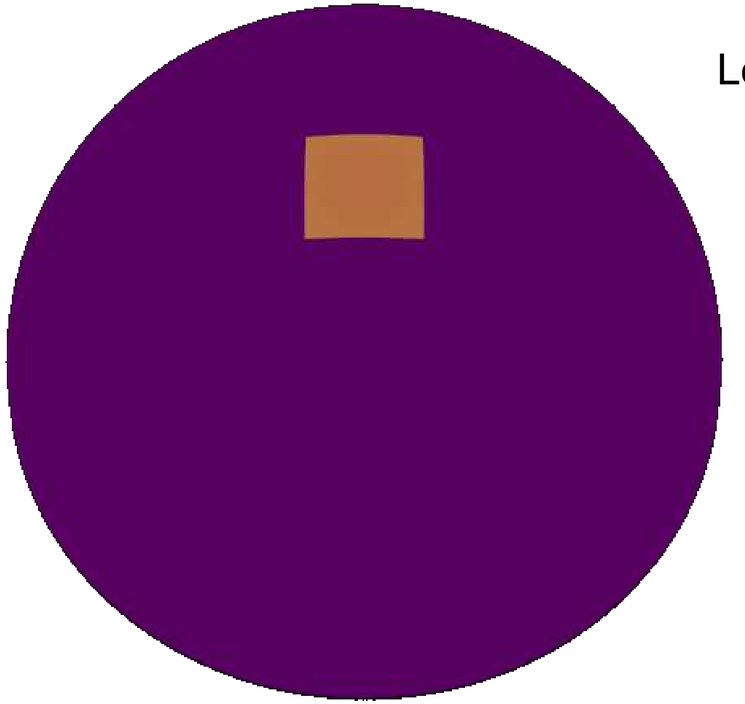
$$I = \int_{\mathbb{H}^2} f(\boldsymbol{\omega_i})\, \mathrm{d}\sigma(\boldsymbol{\omega_i})$$

$$I = \int_{\mathcal{L}(\pi)} f(\boldsymbol{y} \to \boldsymbol{x})\, G(\boldsymbol{x}, \boldsymbol{y})\, \mathrm{d}A(\boldsymbol{y})$$

Includes visibility, fall-off, and foreshortening terms

Reparameterizing the direct illumination integral

Hemispherical integral



Low High

Change of variables

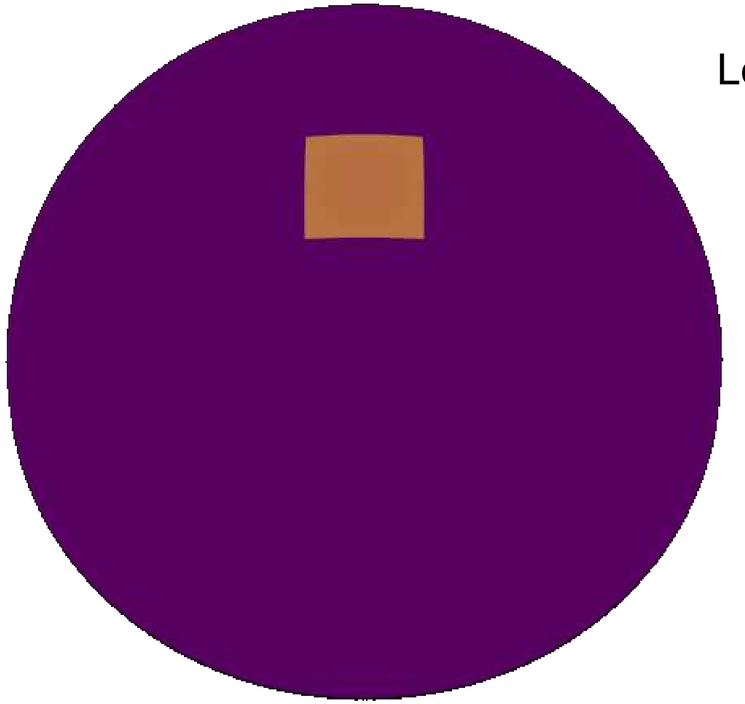
Surface integral

$$I = \int_{\mathbb{H}^2} f(\omega_i)\, \mathrm{d}\sigma(\omega_i)$$

$$I = \int_{\mathcal{L}(\pi)} f(y \to x)\, G(x, y)\, \mathrm{d}A(y)$$

Reparameterizing the direct illumination integral

Hemispherical integral



Low High

Change of variables

Surface integral

$$I = \int_{\mathbb{H}^2} f(\omega_i)\, \mathrm{d}\sigma(\omega_i)$$

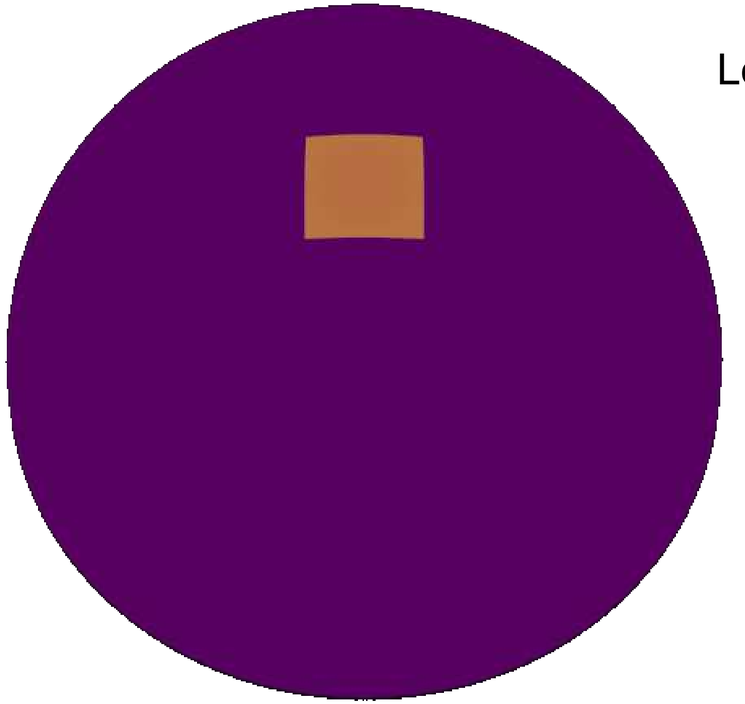
discontinuous

$$I = \int_{\mathcal{L}(\pi)} f(y \to x)\, G(x,y)\, \mathrm{d}A(y)$$

continuous

Reparameterizing the direct illumination integral

Hemispherical integral



Low High

Change of variables

Surface integral

discontinuous

$$I = \int_{\mathbb{H}^2} f(\omega_i)\, \mathrm{d}\sigma(\omega_i)$$

constant domain

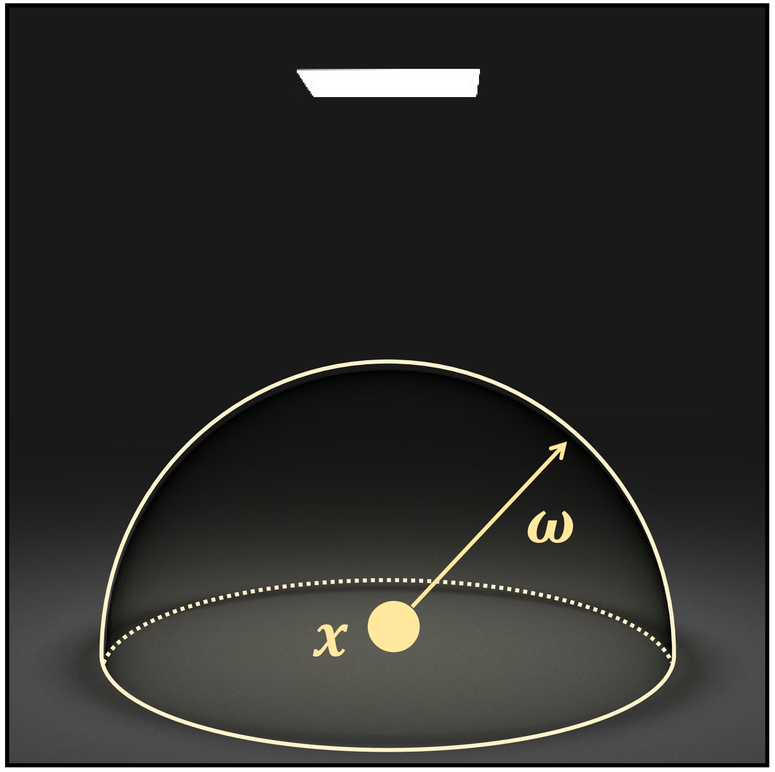
continuous

$$I = \int_{\mathcal{L}(\pi)} f(y \to x)\, G(x, y)\, \mathrm{d}A(y)$$

evolving domain

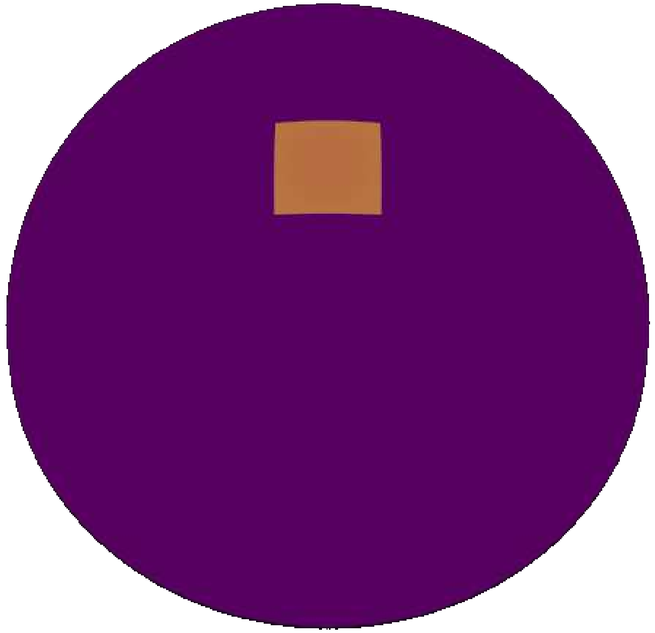
Differentiating the hemispherical integral

π: size of the emitter



$$I = \int_{\mathbb{H}^2} f(\omega_i)\mathrm{d}\sigma(\omega_i)$$

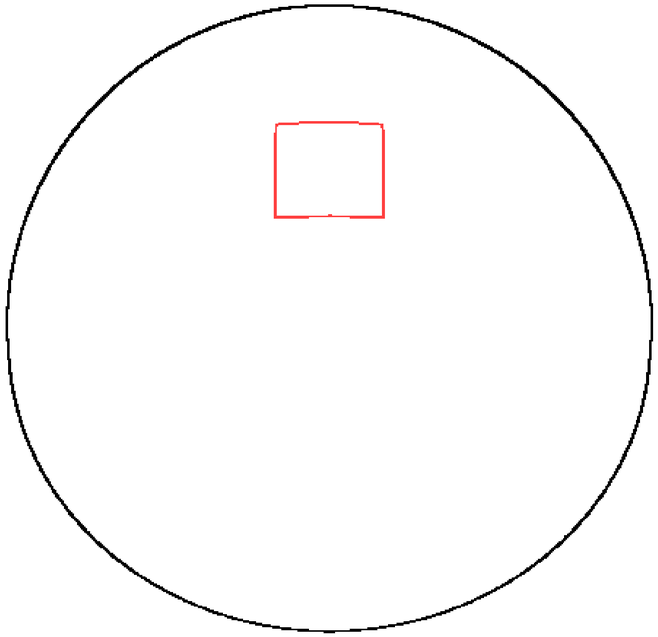
Low — High



Differentiation

Reynolds transport theorem

Discontinuities of f

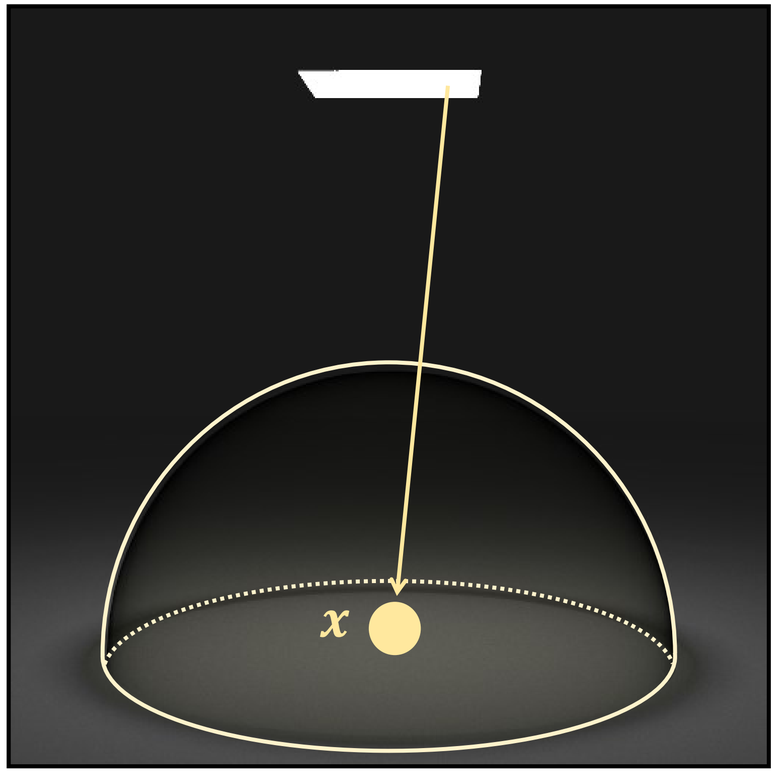


$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathbb{H}^2} \frac{\mathrm{d}(f)}{\mathrm{d}\pi}\mathrm{d}\sigma + \int_{\partial\mathbb{H}^2} g\ \mathrm{d}l$$

Interior — Boundary

Differentiating the area integral

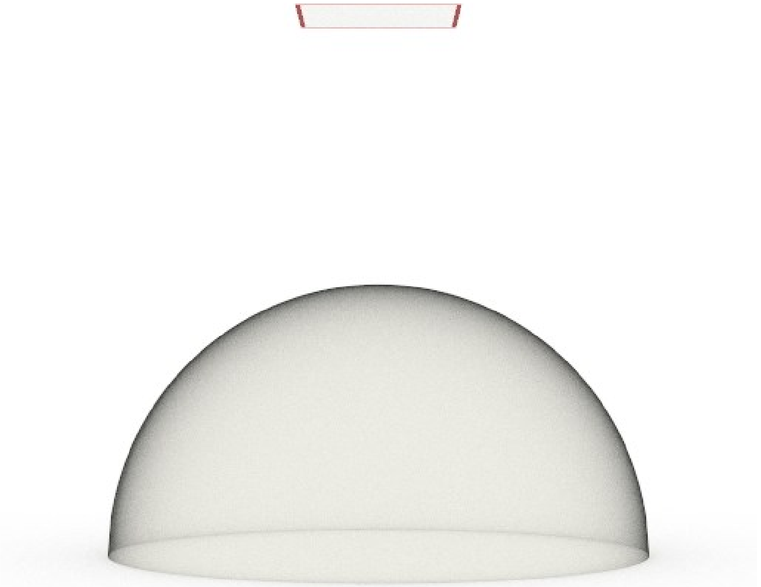
π: size of the emitter



$$I = \int_{\mathcal{L}(\pi)} f(y \to x) G(x, y) \mathrm{d}A(y)$$

Low — High

Boundary of $\mathcal{L}(\pi)$



Differentiation

Reynolds transport theorem

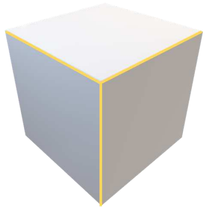
$$\frac{\mathrm{d}I}{\mathrm{d}\pi} = \int_{\mathcal{L}(\pi)} \frac{\mathrm{d}(fG)}{\mathrm{d}\pi} \mathrm{d}A + \int_{\partial\mathcal{L}(\pi)} g \, \mathrm{d}l$$

Interior

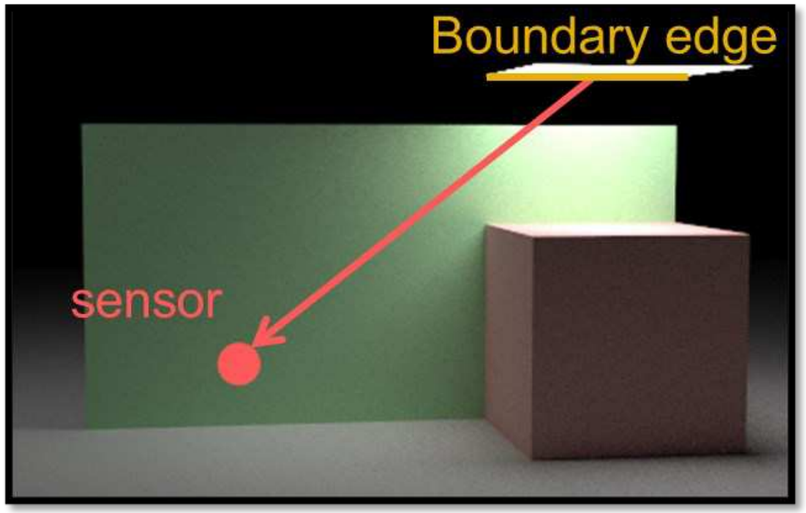
Boundary

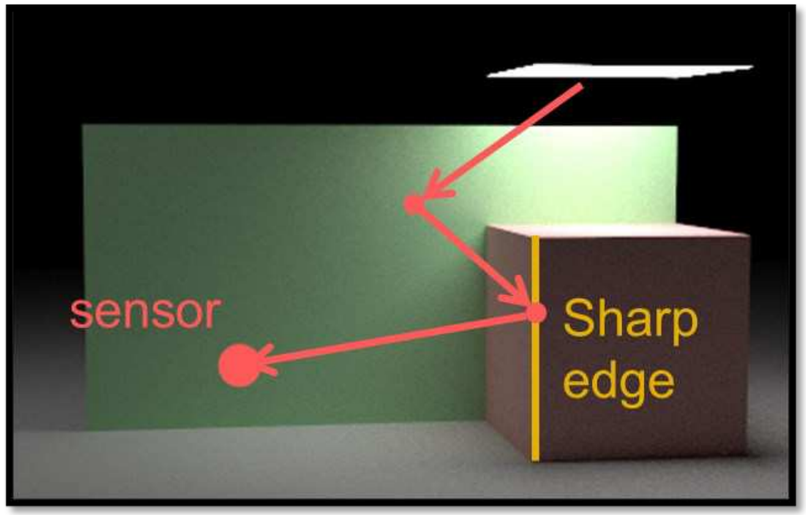
Sources of discontinuities

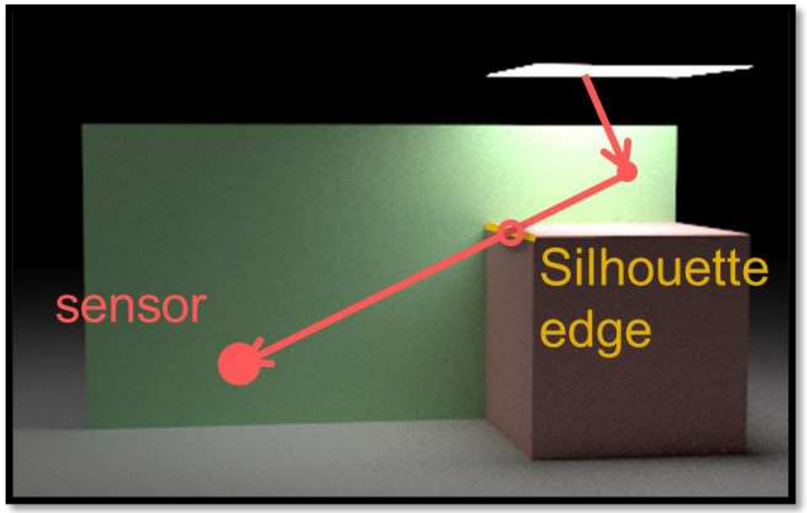
Boundary edge 

Sharp edge 

Silhouette edge 

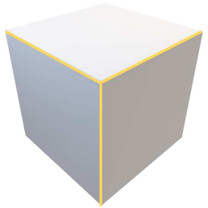




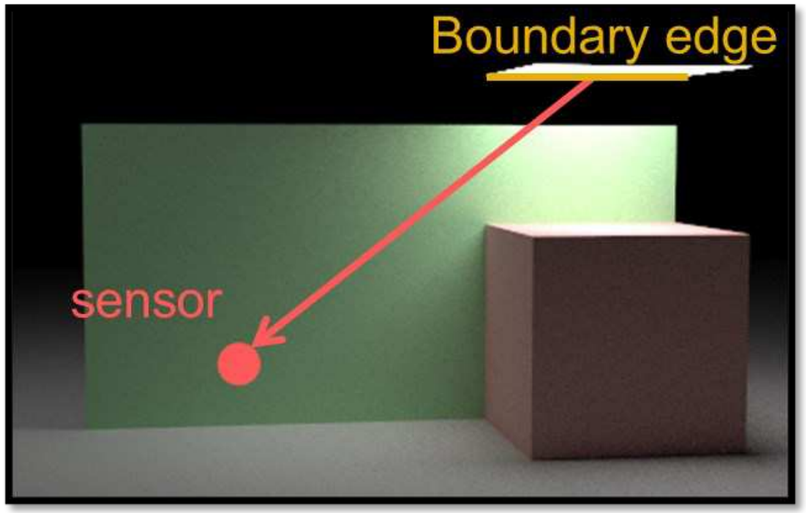


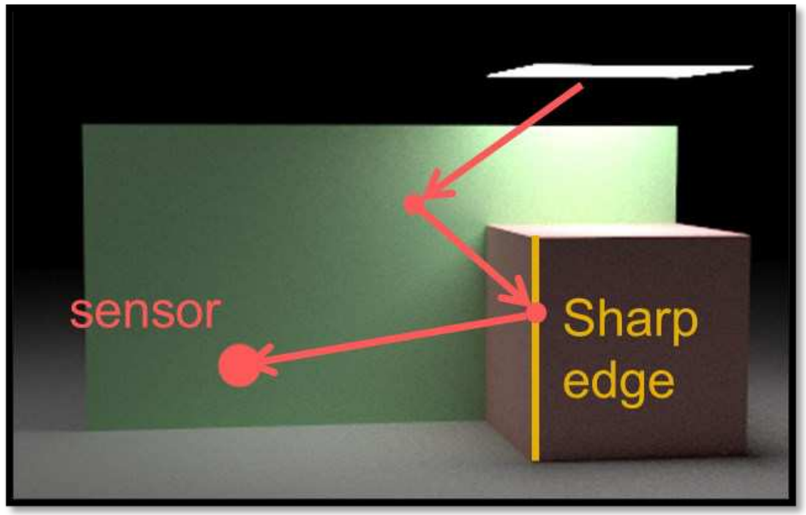
Sources of discontinuities

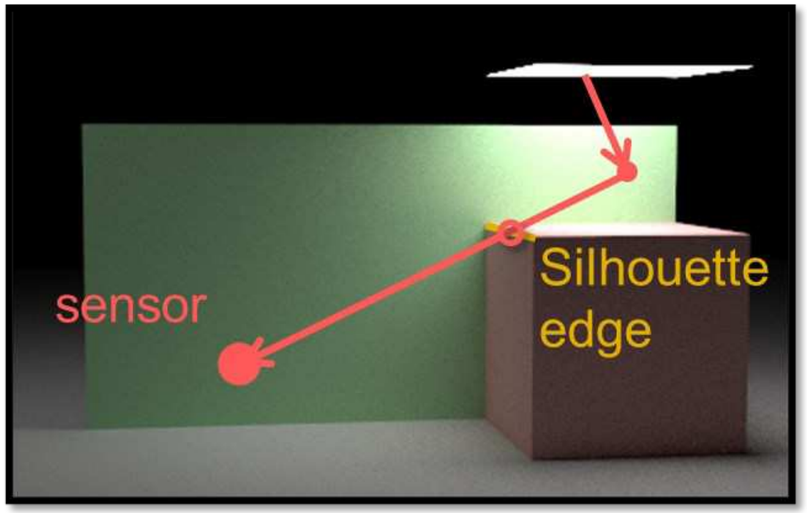
Boundary edge 

Sharp edge 

Silhouette edge 

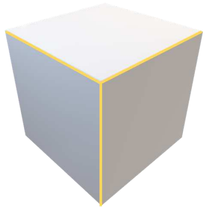




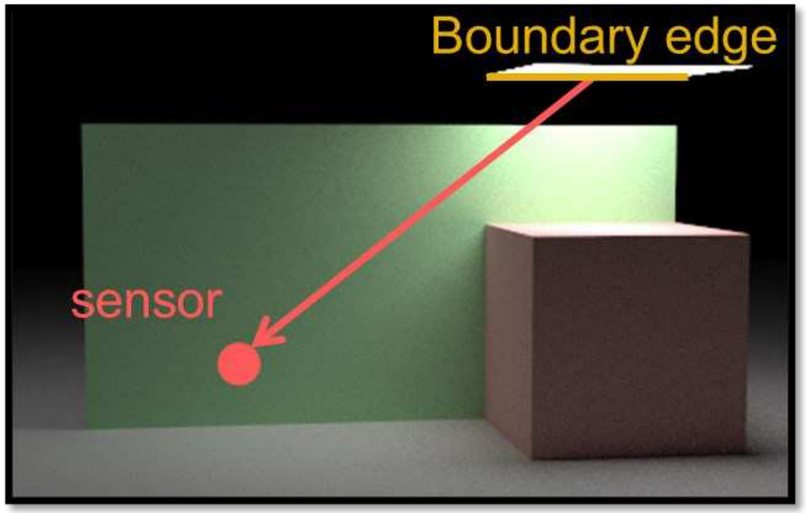


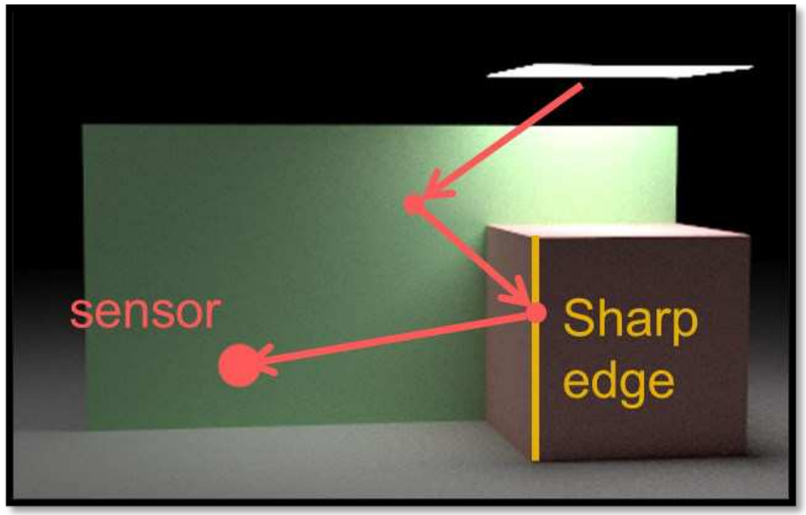
Sources of discontinuities

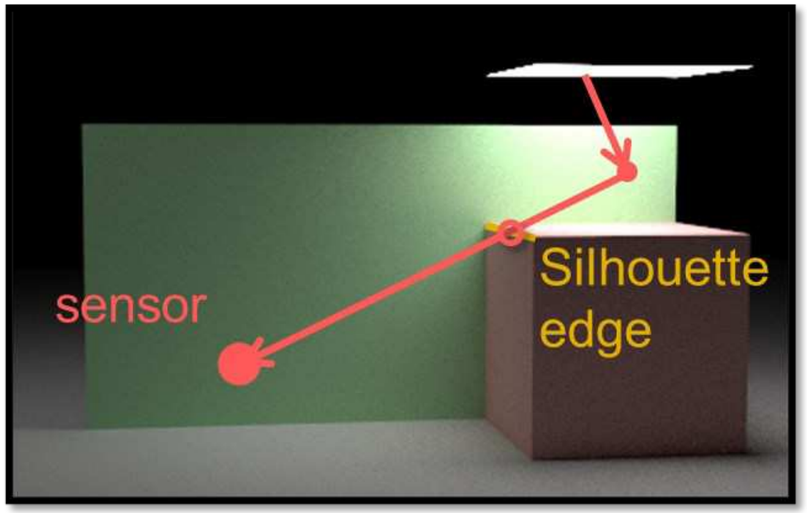
Boundary edge 

Sharp edge 

Silhouette edge 

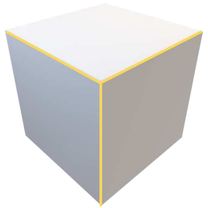




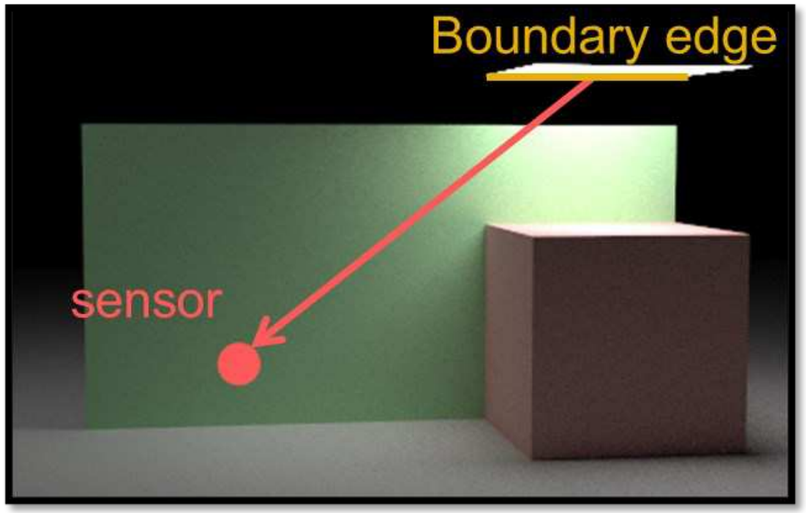


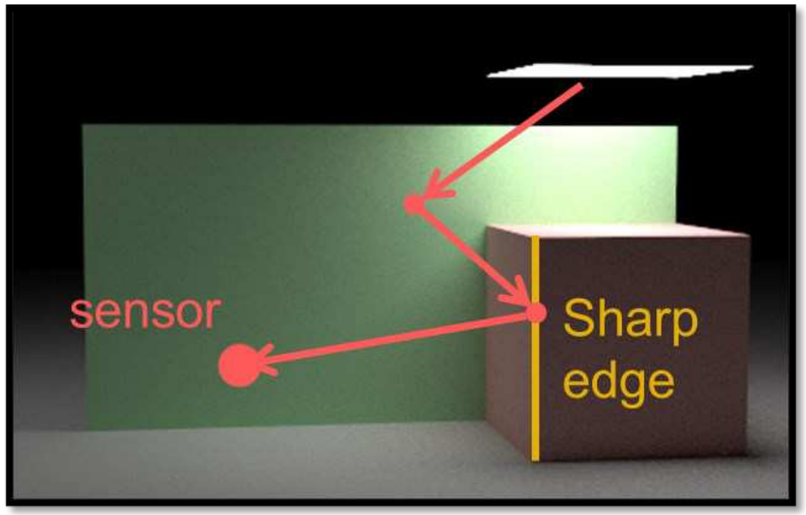
Sources of discontinuities

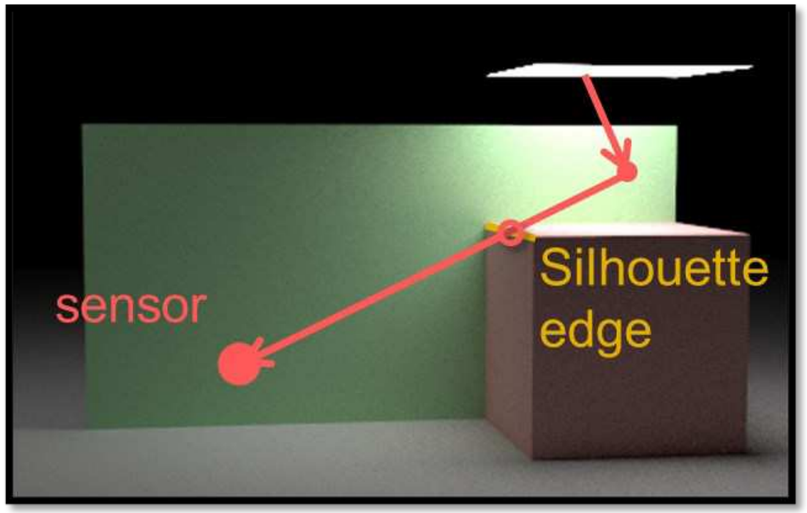
Boundary edge 

Sharp edge 

Silhouette edge 

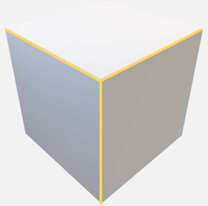




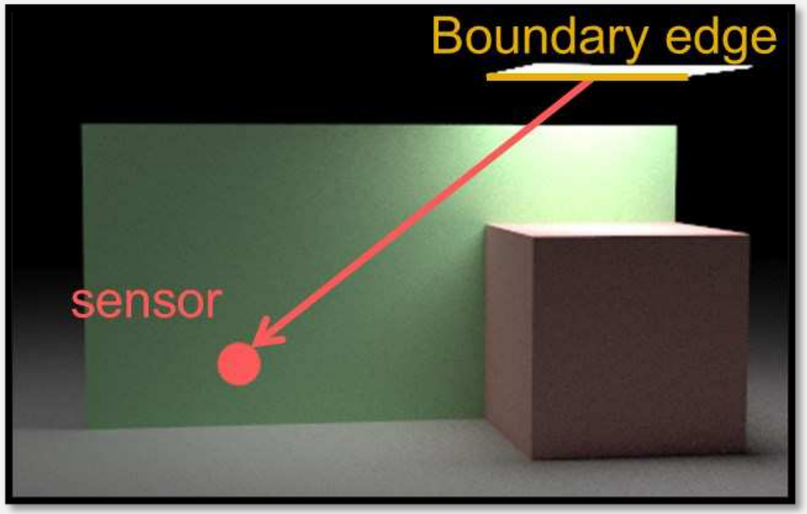


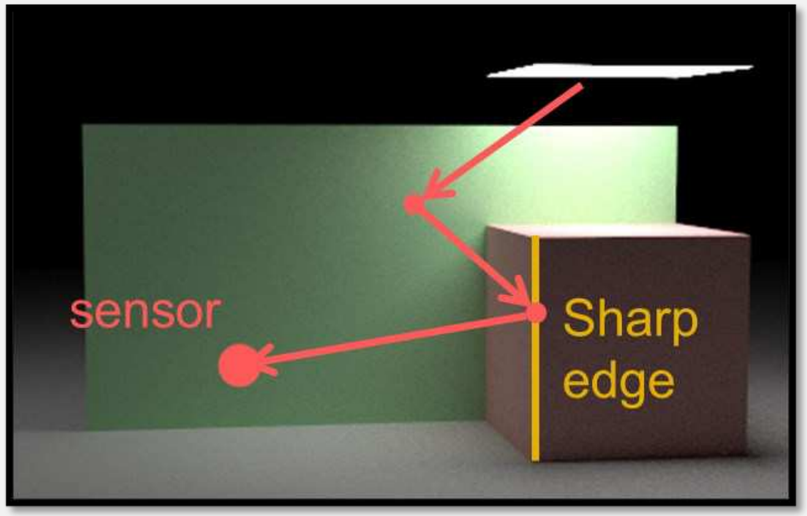
Sources of discontinuities

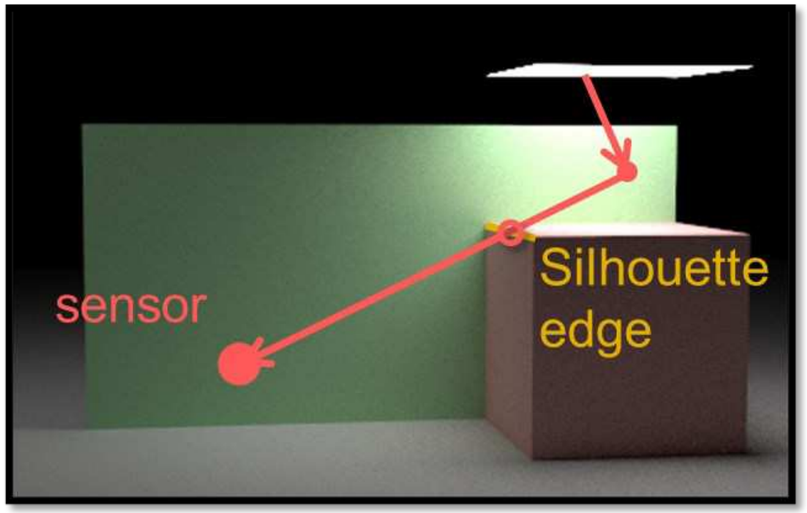
Boundary edge 

Sharp edge 

Silhouette edge 





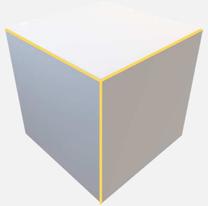


Topology-driven

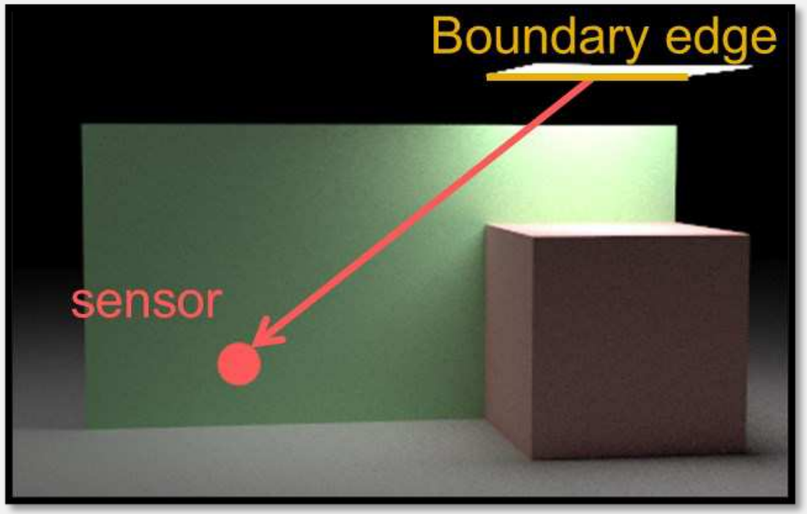
Visibility-driven

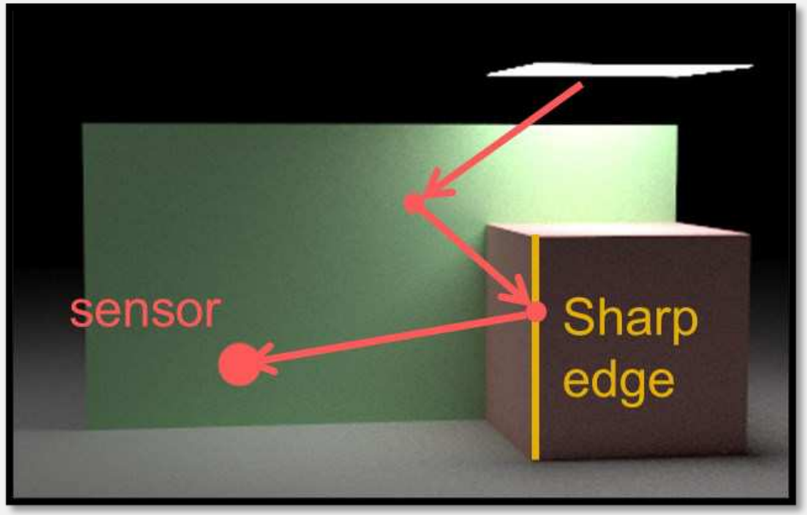
Sources of discontinuities

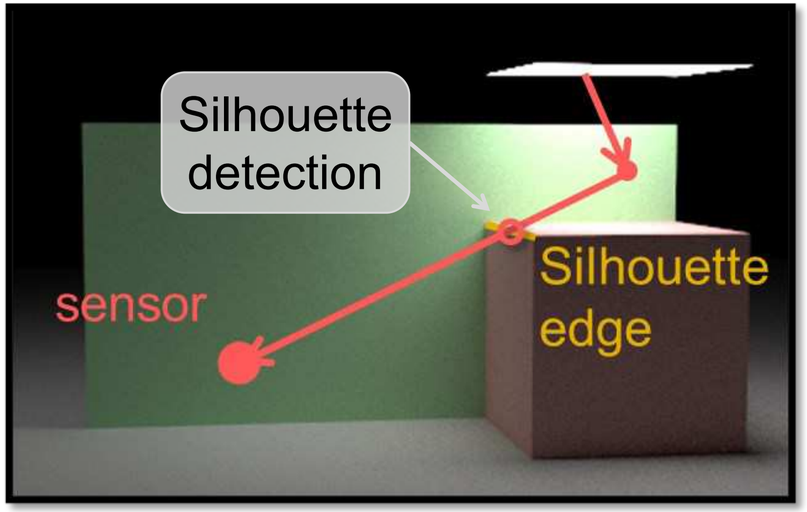
Boundary edge 

Sharp edge 

Silhouette edge 







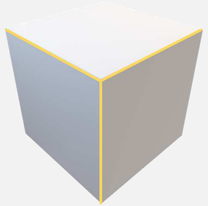
Topology-driven

Visibility-driven

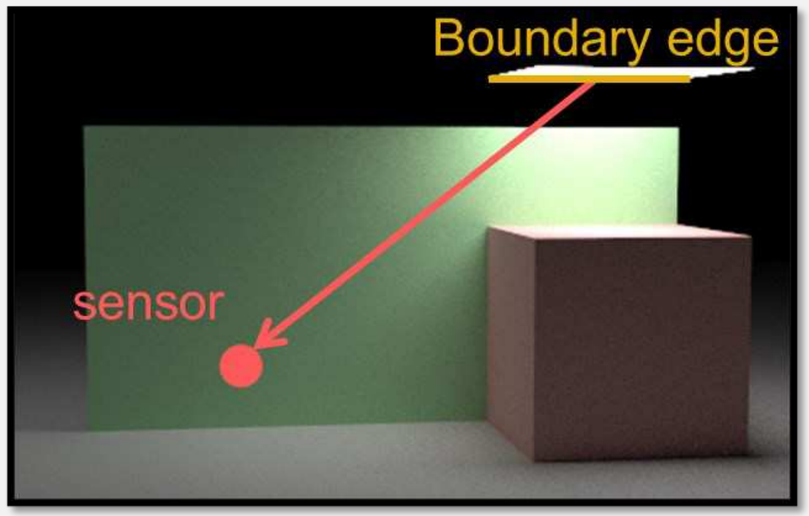
Sources of discontinuities

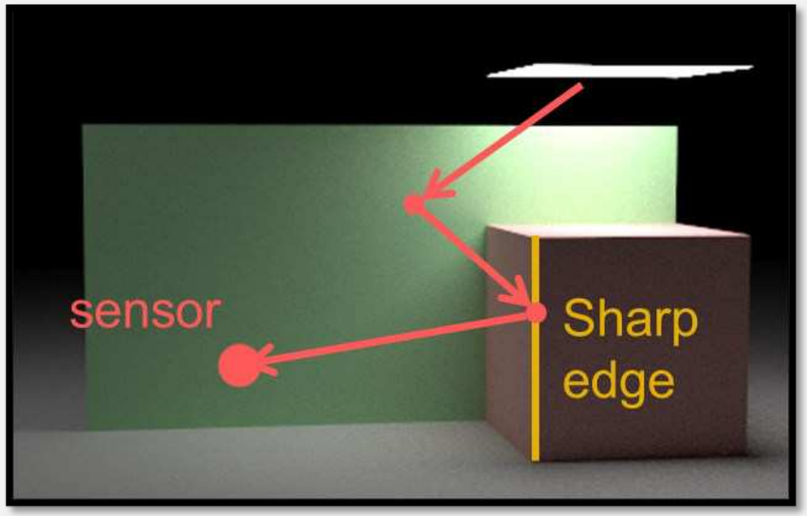
- We still need to account for discontinuities when using smooth closed surfaces (e.g., neural SDFs)

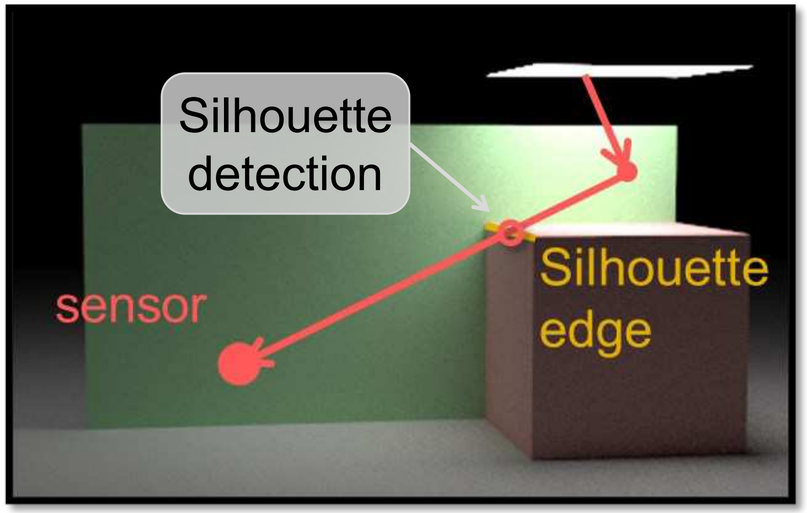
Boundary edge 

Sharp edge 

Silhouette edge 







Topology-driven

Visibility-driven

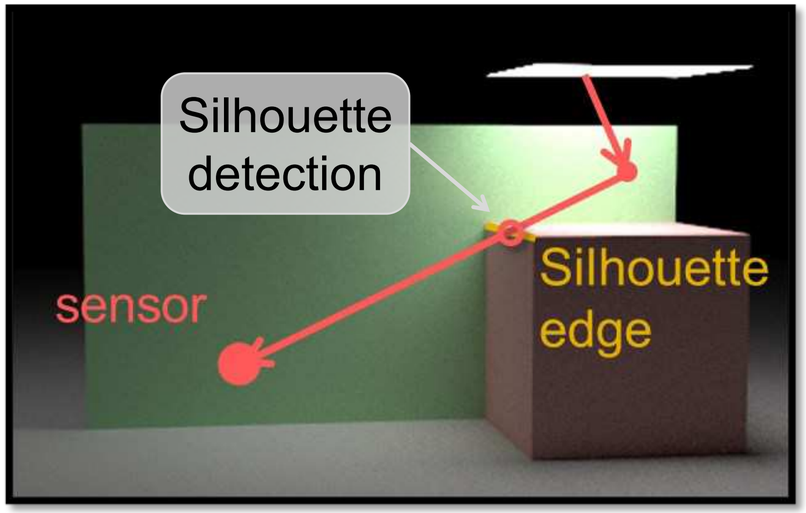
Sources of discontinuities

- We still need to account for discontinuities when using smooth closed surfaces (e.g., neural SDFs)

Boundary edge

Sharp edge

Silhouette edge

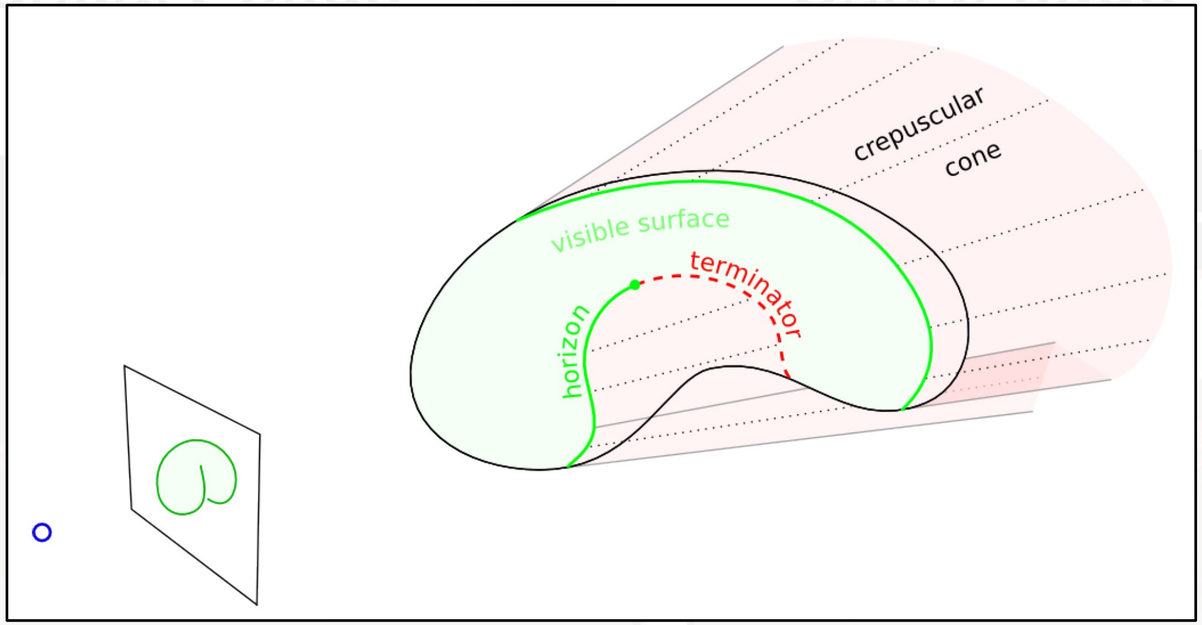


Topology-driven

Visibility-driven

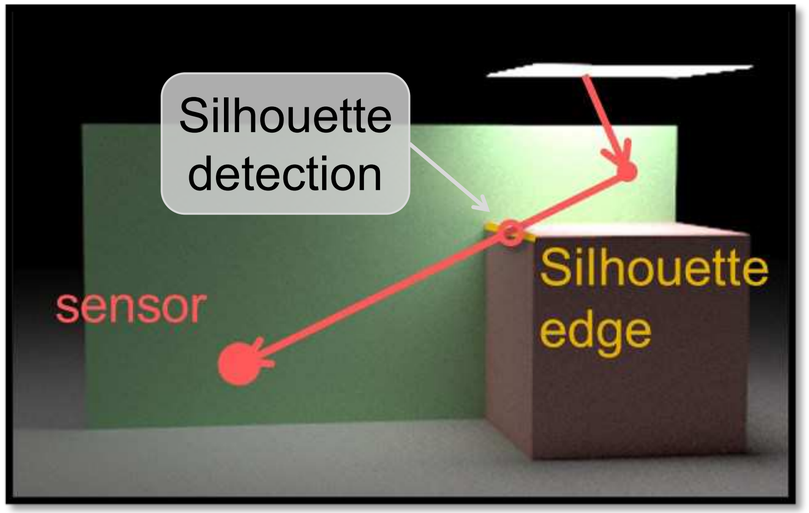
Sources of discontinuities

- We still need to account for discontinuities when using smooth closed surfaces (e.g., neural SDFs)



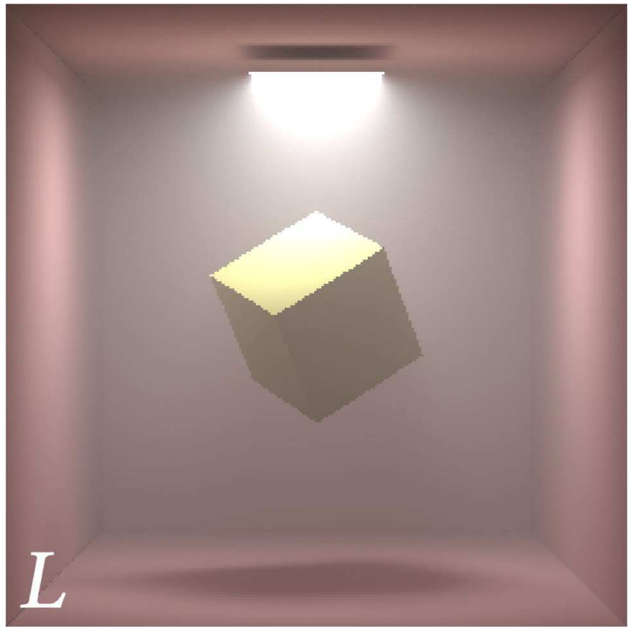
[Gargallo et al., ICCV 2007]

Silhouette edge

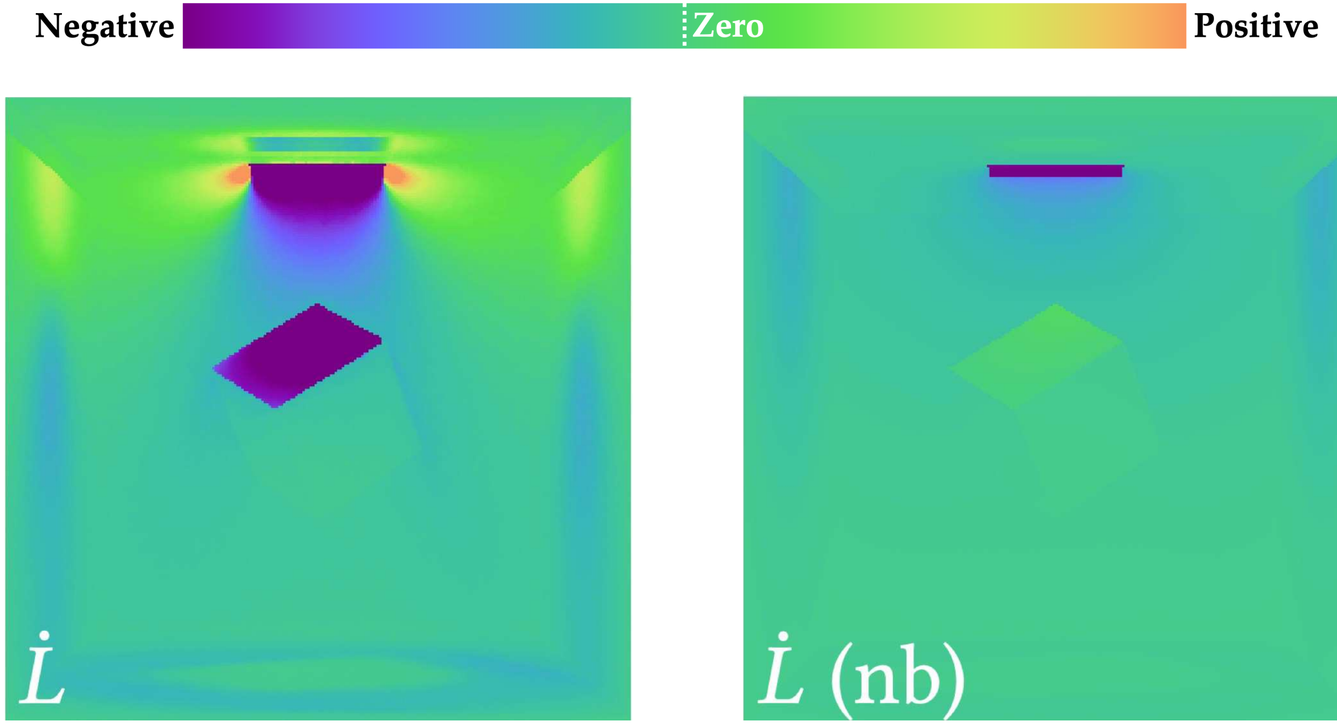


Visibility-driven

Significance of the boundary integral



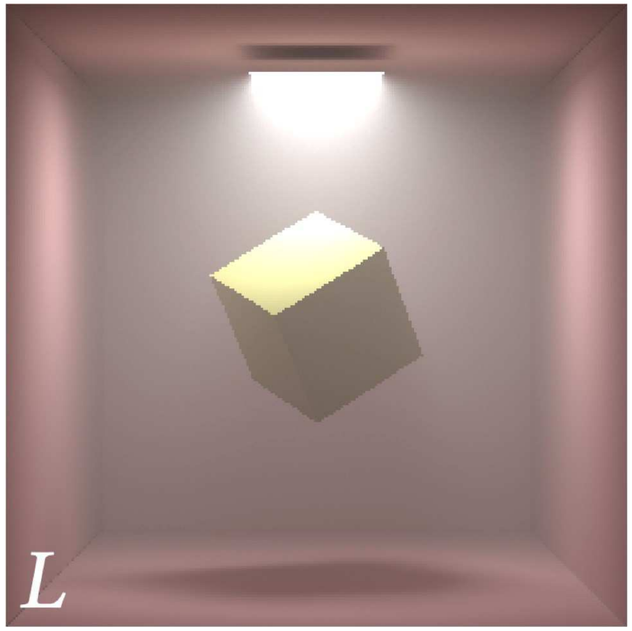
Original image



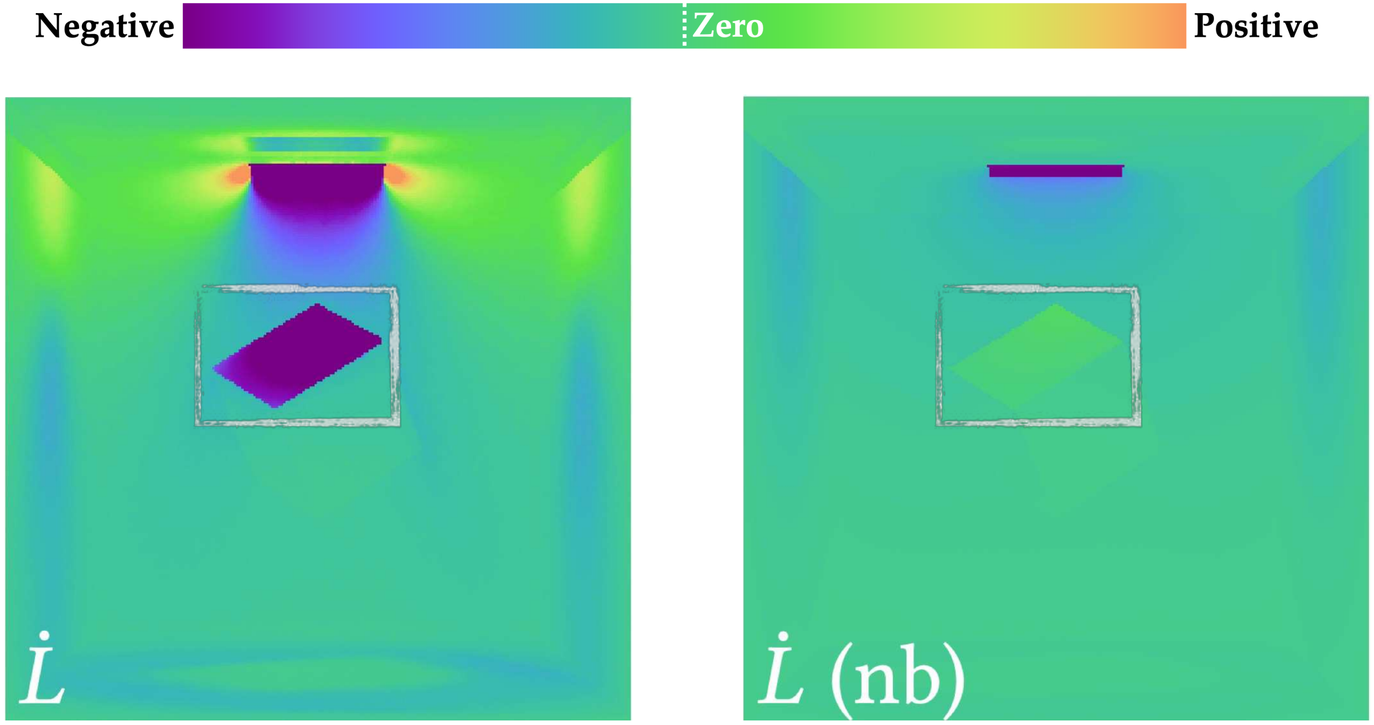
Derivative image
w.r.t. vertical offset of
the area light and the cube

Derivative image
w/o boundary integral

Significance of the boundary integral



Original image

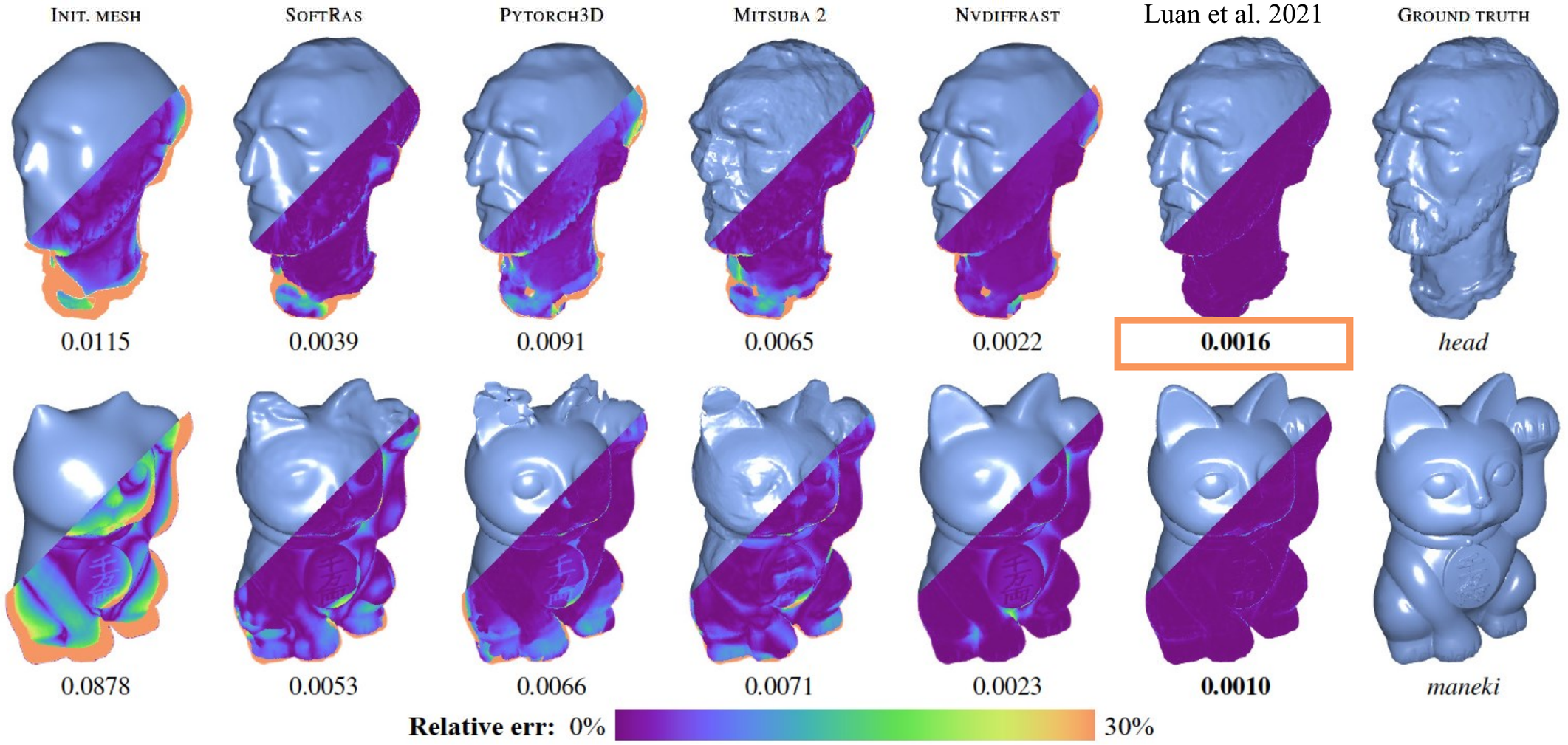


Derivative image
w.r.t. vertical offset of
the area light and the cube

Derivative image
w/o boundary integral

Gradient Accuracy Matters

Inverse-rendering results with *identical* optimization settings



Differential Global Illumination

Very active area of research

Path-Space Differentiable Rendering

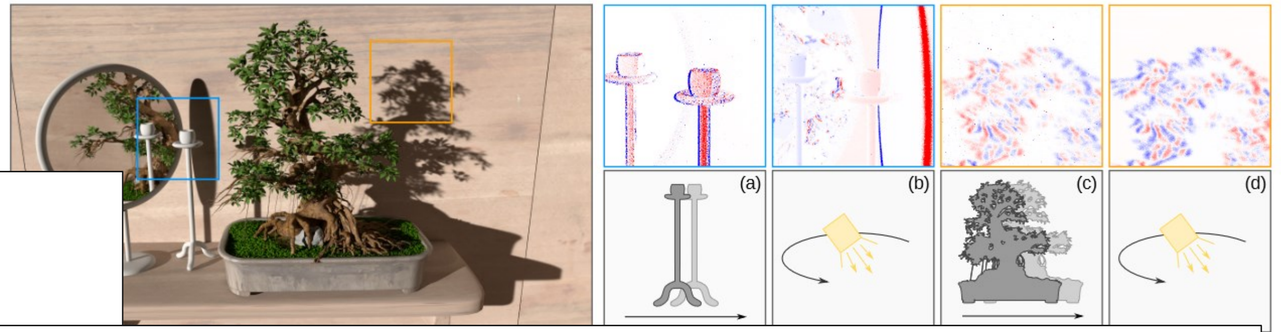
CHENG ZHANG, University of California, Irvine
BAILEY MILLER, Carnegie Mellon University
KAI YAN, University of California, Irvine

Mitsuba 2: A Retargetable Forward and Inverse Renderer

MERLIN NIMIER-DAVID*, École Polytechnique Fédérale de Lausanne
DELIO VICINI*, École Polytechnique Fédérale de Lausanne
TIZIAN ZELTNER, École Polytechnique Fédérale de Lausanne
WENZEL JAKOB, École Polytechnique Fédérale de Lausanne

Reparameterizing Discontinuous Integrands for Differentiable Rendering

GUILLAUME LOUBET, École Polytechnique Fédérale de Lausanne (EPFL)
NICOLAS HOLZSCHUCH, Inria, Univ. Grenoble-Alpes, CNRS, LJK
WENZEL JAKOB, École Polytechnique Fédérale de Lausanne (EPFL)

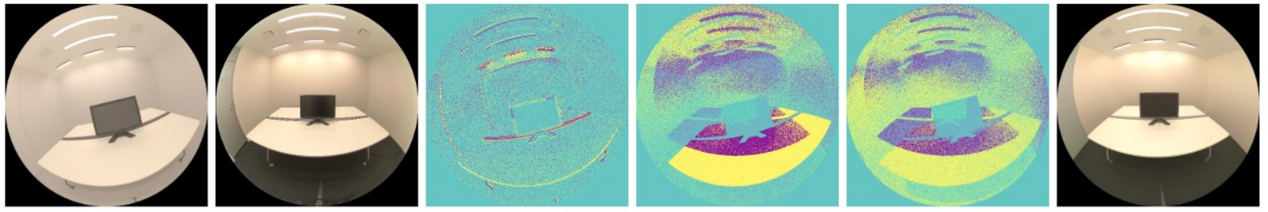


Path Replay Backpropagation: Differentiating Light Paths using Constant Memory and Linear Time

DELIO VICINI, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
SÉBASTIEN SPEIERER, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
WENZEL JAKOB, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Differentiable Monte Carlo Ray Tracing through Edge Sampling

TZU-MAO LI, MIT CSAIL
MIIKA AITTALA, MIT CSAIL
FRÉDO DURAND, MIT CSAIL
JAAKKO LEHTINEN, Aalto University & NVIDIA



(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

Remember: Path Integral for Global Illumination

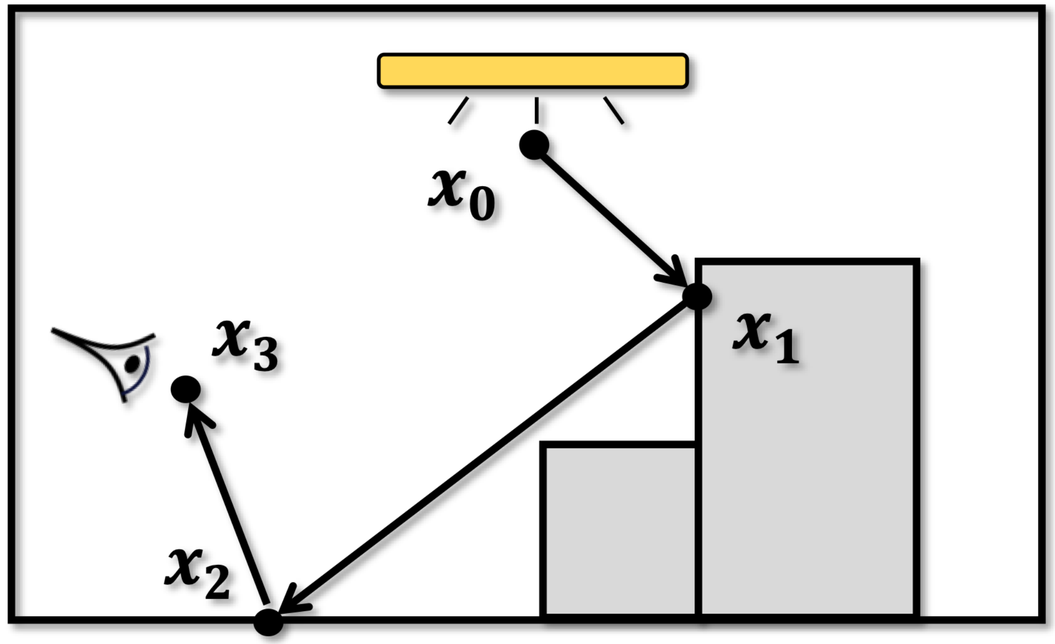
Pixel value

Measurement contribution

$$I = \int_{\Omega} f(\bar{\boldsymbol{x}}) \, \mathrm{d}\, \mu(\bar{\boldsymbol{x}})$$

Path space

Area-product measure



Light path $\bar{\boldsymbol{x}} = (x_0, x_1, x_2, x_3)$

[Zhang et al., 2020]

Differential Path Integral

Path-space differentiable rendering

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\int_\Omega f(\bar{\boldsymbol{x}})\mathrm{d}\mu(\bar{\boldsymbol{x}})\right) = \int_\Omega \dot{f}(\bar{\boldsymbol{x}})\mathrm{d}\mu(\bar{\boldsymbol{x}}) + \int_{\partial\Omega} g(\bar{\boldsymbol{x}})\mathrm{d}\mu'(\bar{\boldsymbol{x}})$$

Interior integral

Boundary integral

[Zhang et al., 2020]

Differential Path Integral

Path-space differentiable rendering

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\int_\Omega f(\overline{\boldsymbol{x}})\,\mathrm{d}\mu(\overline{\boldsymbol{x}})\right) = \int_\Omega \dot{f}(\overline{\boldsymbol{x}})\,\mathrm{d}\mu(\overline{\boldsymbol{x}}) + \int_{\partial\Omega} g(\overline{\boldsymbol{x}})\,\mathrm{d}\mu'(\overline{\boldsymbol{x}})$$

Interior integral **Boundary** integral

We now derive $\partial I_N/\partial\pi$ in Eq. (25) using the recursive relations provided by Eqs. (21) and (24). Let

$$h_n^{(0)} \coloneqq \left[\prod_{n'=n+1}^{N} g(\boldsymbol{x}_{n'};\, \boldsymbol{x}_{n'-2}, \boldsymbol{x}_{n'-1})\right] W_\mathrm{e}(\boldsymbol{x}_N \to \boldsymbol{x}_{N-1}), \quad (52)$$

$$h_n^{(1)} \coloneqq \sum_{n'=n+1}^{N} \kappa(\boldsymbol{x}_{n'})\, V(\boldsymbol{x}_{n'}), \quad (53)$$

$$\Delta h_{n,n'}^{(0)} \coloneqq h_n^{(0)}\, \Delta g(\boldsymbol{x}_{n'};\boldsymbol{x}_{n'-2},\boldsymbol{x}_{n'-1}) / g(\boldsymbol{x}_{n'};\boldsymbol{x}_{n'-2},\boldsymbol{x}_{n'-1}), \quad (54)$$

for $0 \le n < n' \le N$. We omit the dependencies of $h_n^{(0)}$, $h_n^{(1)}$, and $\Delta h_{n,n'}^{(0)}$ on $\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N$ for notational convenience.

We now show that, for all $0 \le n < N$, it holds that

$$h_n(\boldsymbol{x}_n;\, \boldsymbol{x}_{n-1}) = \int_{\mathcal{M}^{N-n}} h_n^{(0)} \prod_{n'=n+1}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}), \quad (55)$$

and

$$\dot{h}_n(\boldsymbol{x}_n;\, \boldsymbol{x}_{n-1}) = \int_{\mathcal{M}^{N-n}} \left[\left(h_n^{(0)}\right)^{\cdot} - h_n^{(0)} h_n^{(1)}\right] \prod_{n'=n+1}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}) + \sum_{n'=n+1}^{N} \int \Delta h_{n,n'}^{(0)}\, V_{\overline{\partial\mathcal{M}}_{n'}}(\boldsymbol{x}_{n'})\, \mathrm{d}\ell(\boldsymbol{x}_{n'}) \prod_{\substack{n<i\le N \\ i\ne n'}} \mathrm{d}A(\boldsymbol{x}_i), \quad (56)$$

where the integral domain of the second term on the right-hand side, which is omitted for notational clarity, is $\mathcal{M}(\pi)$ for each $\boldsymbol{x}_i$ with $i \ne n'$ and $\overline{\partial\mathcal{M}}_{n'}(\pi)$, which depends on $\boldsymbol{x}_{n'-1}$, for $\boldsymbol{x}_{n'}$.

It is easy to verify that Eqs. (55) and (56) hold for $n = N-1$. We now show that, if they hold for some $0 < n < N$, then it is also the case for $n-1$. Let $g_{n-1} \coloneqq g(\boldsymbol{x}_n;\, \boldsymbol{x}_{n-2}, \boldsymbol{x}_{n-1})$ for all $0 < n \le N$. Then,

$$h_{n-1}(\boldsymbol{x}_{n-1};\, \boldsymbol{x}_{n-2}) = \int_{\mathcal{M}} g_{n-1} \int_{\mathcal{M}^{N-n}} h_n^{(0)} \prod_{n'=n+1}^{N} \mathrm{d}A(\boldsymbol{x}_{n'})\, \mathrm{d}A(\boldsymbol{x}_n) = \int_{\mathcal{M}^{N-n+1}} h_{n-1}^{(0)} \prod_{n'=n}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}), \quad (57)$$

and

$$\begin{aligned}
&\dot{h}_{n-1}(\boldsymbol{x}_{n-1};\, \boldsymbol{x}_{n-2}) \\
&= \int_{\mathcal{M}} \left[\dot{g}_{n-1}\, h_n + g_{n-1}(\dot{h}_n - h_n\, \kappa(\boldsymbol{x}_n)\, V(\boldsymbol{x}_n))\right] \mathrm{d}A(\boldsymbol{x}_n) \\
&\quad + \int_{\overline{\partial\mathcal{M}}_n} \Delta g_{n-1}\, h_n\, V_{\overline{\partial\mathcal{M}}_n}\, \mathrm{d}\ell(\boldsymbol{x}_n) \\
&= \int_{\mathcal{M}^{N-n+1}} \left\{\dot{g}_{n-1}\, h_n^{(0)} + g_{n-1}\left[\left(h_n^{(0)}\right)^{\cdot} - h_n^{(0)} h_{n-1}^{(1)}\right]\right\} \prod_{n'=k}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}) \\
&\quad + \sum_{n'=n+1}^{N} \int g_{n-1}\, \Delta h_{n,n'}^{(0)}\, V_{\overline{\partial\mathcal{M}}_{n'}}(\boldsymbol{x}_{n'})\, \mathrm{d}\ell(\boldsymbol{x}_{n'}) \prod_{\substack{n\le i\le N \\ i\ne n'}} \mathrm{d}A(\boldsymbol{x}_i) \\
&\quad + \int \Delta g_{n-1}\, h_n^{(0)}\, V_{\overline{\partial\mathcal{M}}_n}\, \mathrm{d}\ell(\boldsymbol{x}_n) \prod_{n'=n+1}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}) \\
&= \int_{\mathcal{M}^{N-n+1}} \left[\left(h_{n-1}^{(0)}\right)^{\cdot} - h_{n-1}^{(0)} h_{n-1}^{(1)}\right] \prod_{n'=n}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}) \\
&\quad + \sum_{n'=n}^{N} \int \Delta h_{n-1,n'}^{(0)}\, V_{\overline{\partial\mathcal{M}}_{n'}}(\boldsymbol{x}_{n'})\, \mathrm{d}\ell(\boldsymbol{x}_{n'}) \prod_{\substack{n\le i\le N \\ i\ne n'}} \mathrm{d}A(\boldsymbol{x}_i). \quad (58)
\end{aligned}$$

Thus, using mathematical induction, we know that Eqs. (55) and (56) hold for all $0 \le n < N$.

Notice that $h_0^{(0)} = f$ and $\Delta h_{0,n'}^{(0)} = \Delta f_{n'}$, where $\Delta f_{n'}$ follows the definition in Eq. (28). Letting $n = 0$ in Eq. (56) yields

$$\dot{h}_0(\boldsymbol{x}_0) = \int_{\mathcal{M}^N} \left[\dot{f}(\tilde{\boldsymbol{x}}) - f(\tilde{\boldsymbol{x}}) \sum_{n'=1}^{N} \kappa(\boldsymbol{x}_{n'})\, V(\boldsymbol{x}_{n'})\right] \prod_{n'=1}^{N} \mathrm{d}A(\boldsymbol{x}_{n'}) + \sum_{n'=1}^{N} \int \Delta f_{n'}(\tilde{\boldsymbol{x}})\, V_{\overline{\partial\mathcal{M}}_{n'}}\, \mathrm{d}\ell(\boldsymbol{x}_{n'}) \prod_{\substack{0<i\le N \\ i\ne n'}} \mathrm{d}A(\boldsymbol{x}_i). \quad (59)$$

Lastly, based on the assumption that h_0 is continuous in $\boldsymbol{x}_0$, Eq. (25) can be obtained by differentiating Eq. (23):

$$\begin{aligned}
\frac{\partial I_N}{\partial\pi} &= \frac{\partial}{\partial\pi} \int_{\mathcal{M}} h_0(\boldsymbol{x}_0)\, \mathrm{d}A(\boldsymbol{x}_0) \\
&= \int_{\mathcal{M}} \left[\dot{h}_0(\boldsymbol{x}_0) - h_0(\boldsymbol{x}_0)\, \kappa(\boldsymbol{x}_0)\, V(\boldsymbol{x}_0)\right] \mathrm{d}A(\boldsymbol{x}_0) \\
&\quad + \int_{\overline{\partial\mathcal{M}}_0} h_0(\boldsymbol{x}_0)\, V_{\overline{\partial\mathcal{M}}_0}(\boldsymbol{x}_0)\, \mathrm{d}\ell(\boldsymbol{x}_0) \\
&= \int_{\Omega_N} \left[\dot{f}(\tilde{\boldsymbol{x}}) - f(\tilde{\boldsymbol{x}}) \sum_{K=0}^{N} \kappa(\boldsymbol{x}_K)\, V(\boldsymbol{x}_K)\right] \mathrm{d}\mu(\tilde{\boldsymbol{x}}) \\
&\quad + \sum_{K=0}^{N} \int_{\Omega_{N,K}} \Delta f_K(\tilde{\boldsymbol{x}})\, V_{\overline{\partial\mathcal{M}}_K}\, \mathrm{d}\mu'_{N,K}(\tilde{\boldsymbol{x}}).
\end{aligned} \quad (60)$$

(The full derivation is quite involved…)

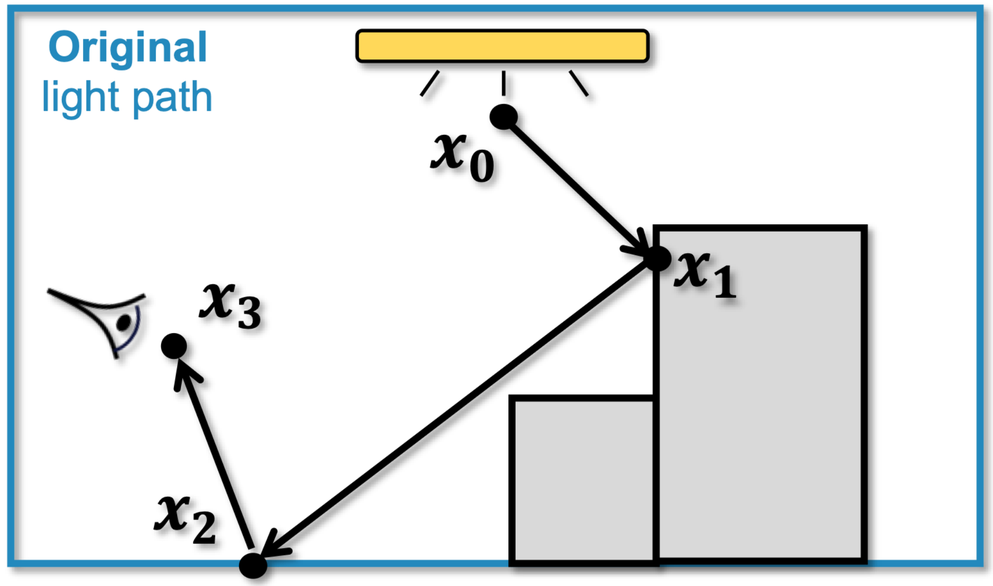
[Zhang et al., 2020]

Differential Path Integral

Path-space differentiable rendering

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\int_\Omega f(\overline{\boldsymbol{x}})\mathrm{d}\mu(\overline{\boldsymbol{x}})\right) = \int_\Omega \dot{f}(\overline{\boldsymbol{x}})\mathrm{d}\mu(\overline{\boldsymbol{x}}) + \int_{\partial\Omega} g(\overline{\boldsymbol{x}})\mathrm{d}\mu'(\overline{\boldsymbol{x}})$$

Interior integral



Interior integral

- Defined on the ordinary path space Ω
- The integrand $\dot{f}$ can be obtained by differentiating the ordinary *measurement contribution function* f

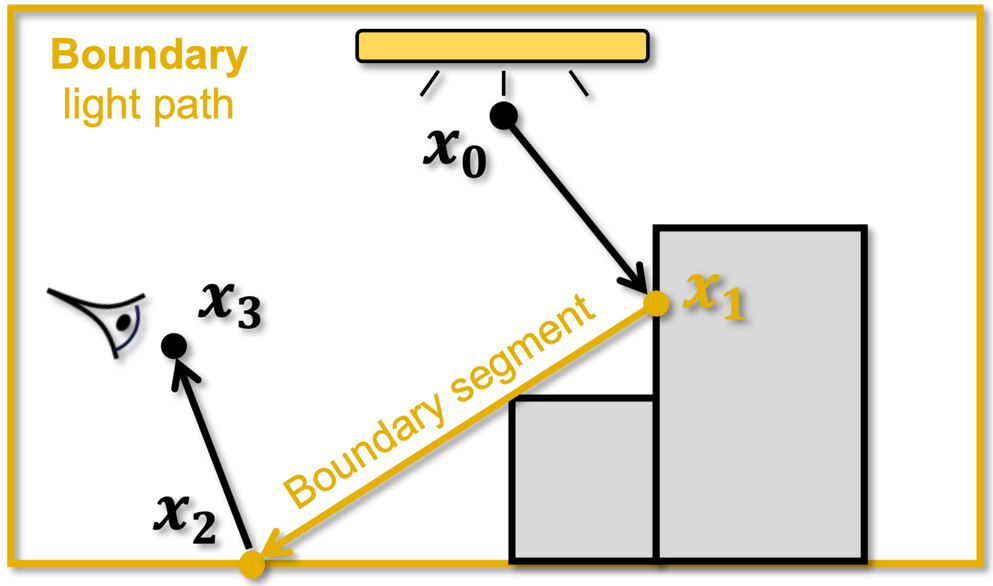
[Zhang et al., 2020]

Differential Path Integral

Path-space differentiable rendering

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\int_\Omega f(\bar{\boldsymbol{x}})\mathrm{d}\mu(\bar{\boldsymbol{x}})\right) = \int_\Omega \dot{f}(\bar{\boldsymbol{x}})\mathrm{d}\mu(\bar{\boldsymbol{x}}) + \int_{\partial\Omega} g(\bar{\boldsymbol{x}})\mathrm{d}\mu'(\bar{\boldsymbol{x}})$$

Boundary integral



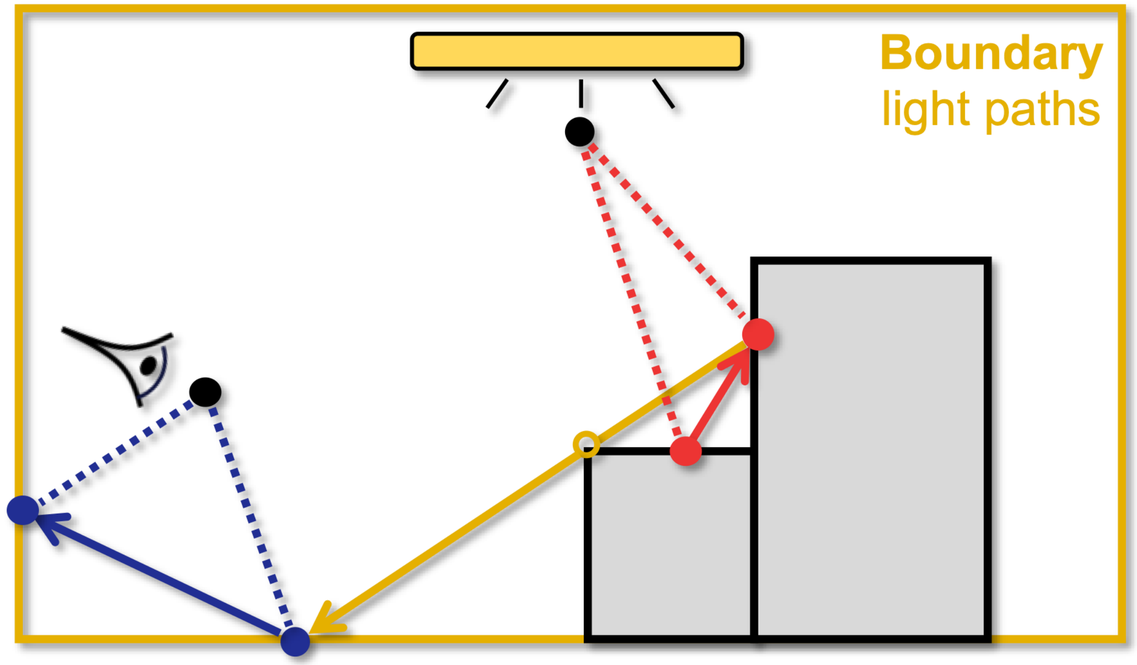
Boundary integral

- Defined on the boundary path space $\partial\Omega$
- A **boundary** light path is the same as an original one except having exactly one boundary segment

Path-Space Differentiable Path Tracing

Unidirectional estimator

- **Interior**: *unidirectional* path tracing
- **Boundary**: *unidirectional* sampling of subpaths



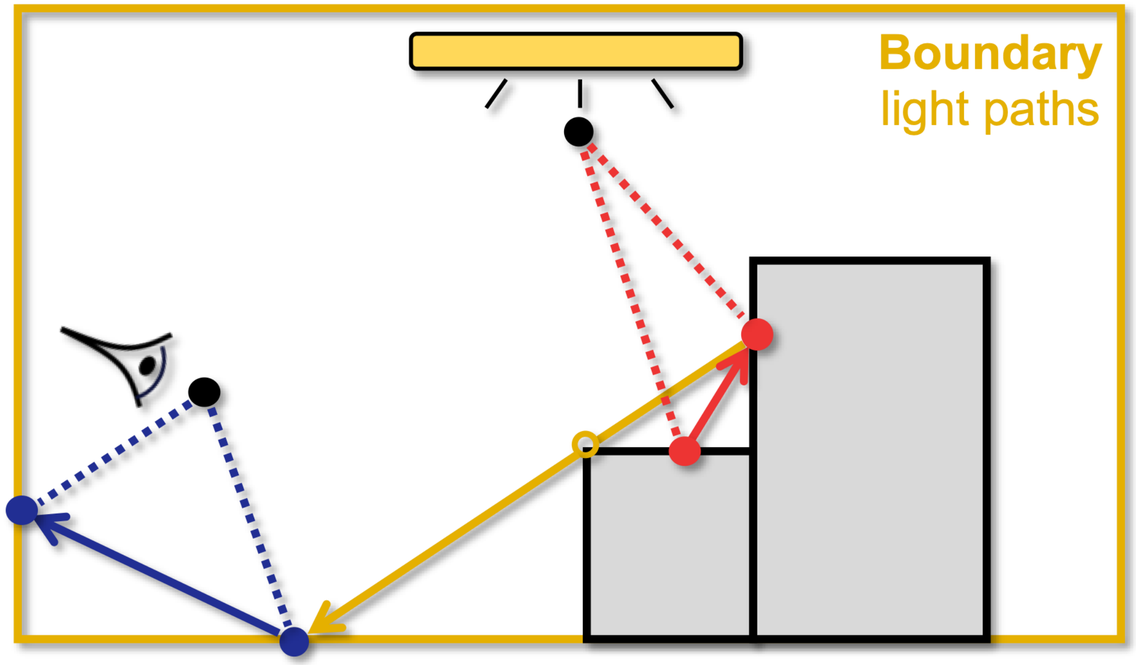
Unidirectional path tracing + NEE

[Zhang et al., 2020]

Path-Space Differentiable Path Tracing

Unidirectional estimator

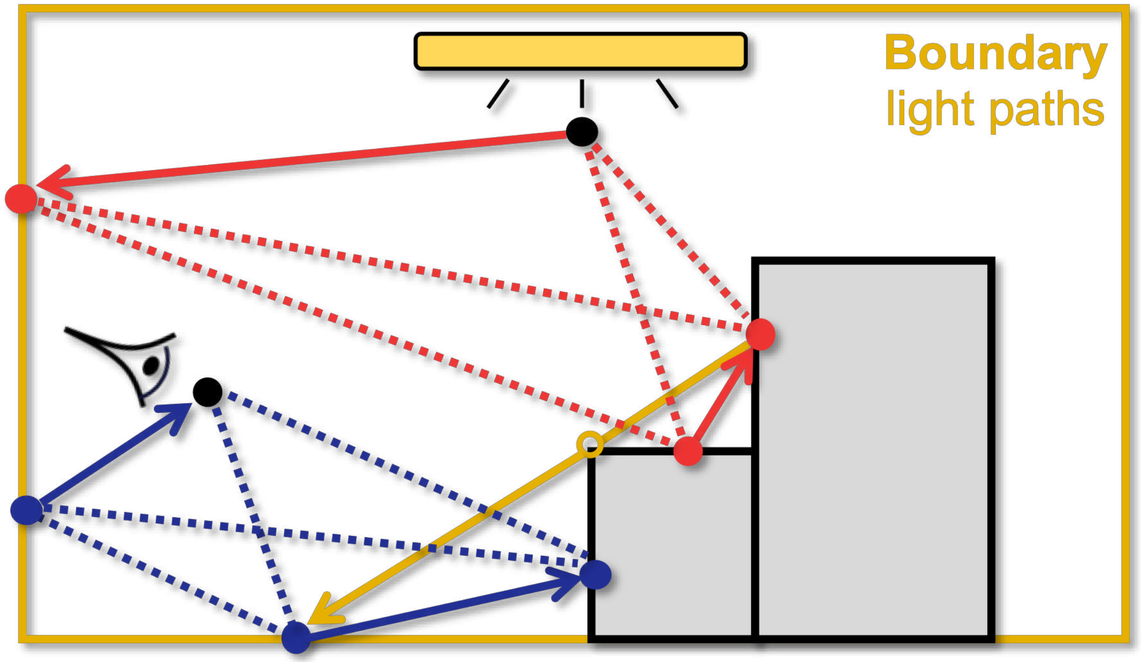
- **Interior**: *unidirectional* path tracing
- **Boundary**: *unidirectional* sampling of subpaths



Unidirectional path tracing + NEE

Bidirectional estimator

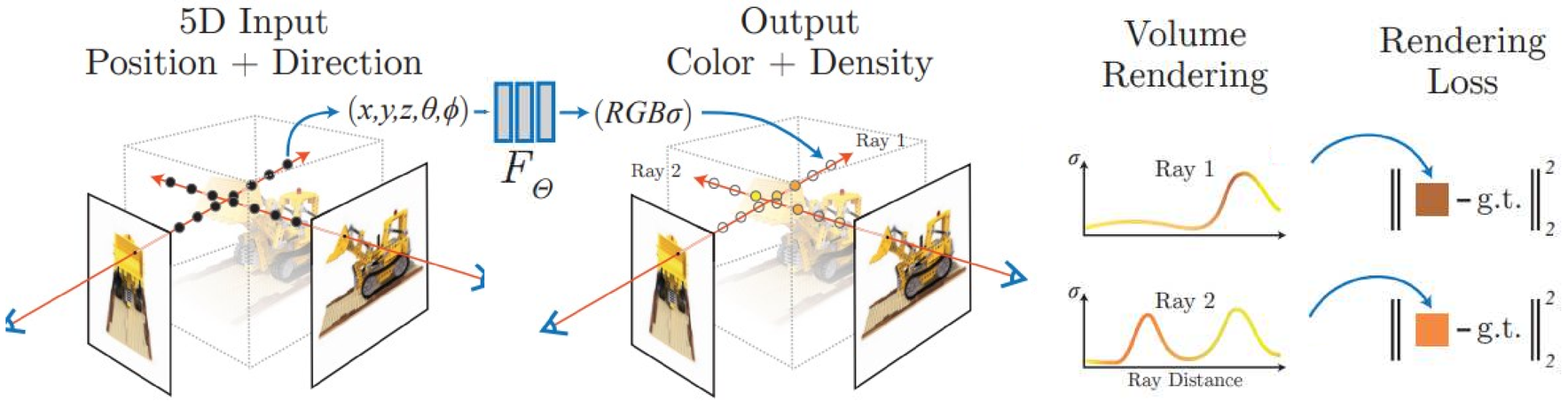
- **Interior**: *bidirectional* path tracing
- **Boundary**: *bidirectional* sampling of subpaths



Bidirectional path tracing

[Zhang et al., 2020]

Application: neural rendering

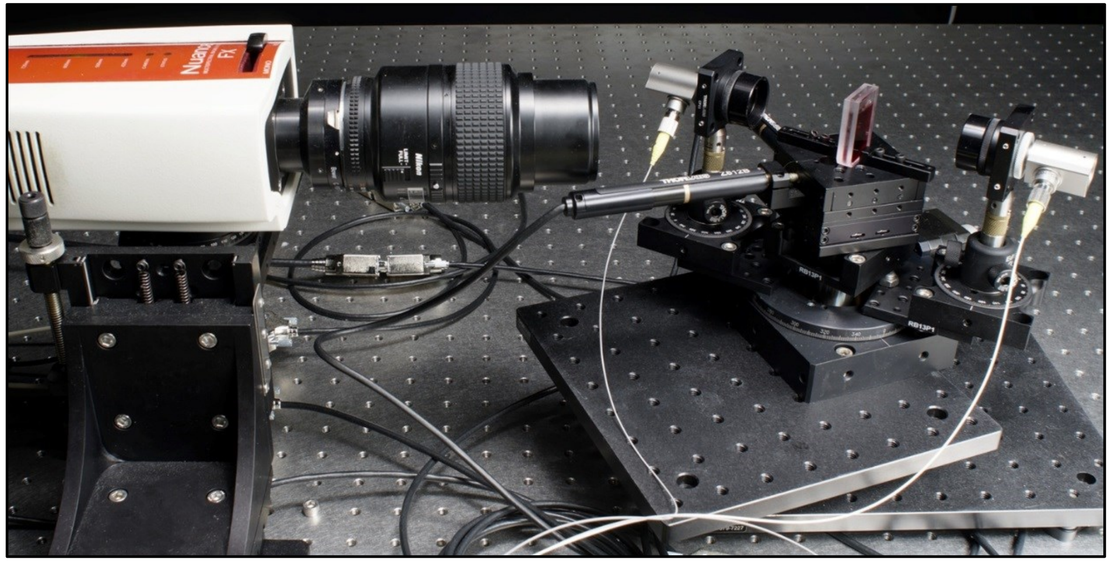


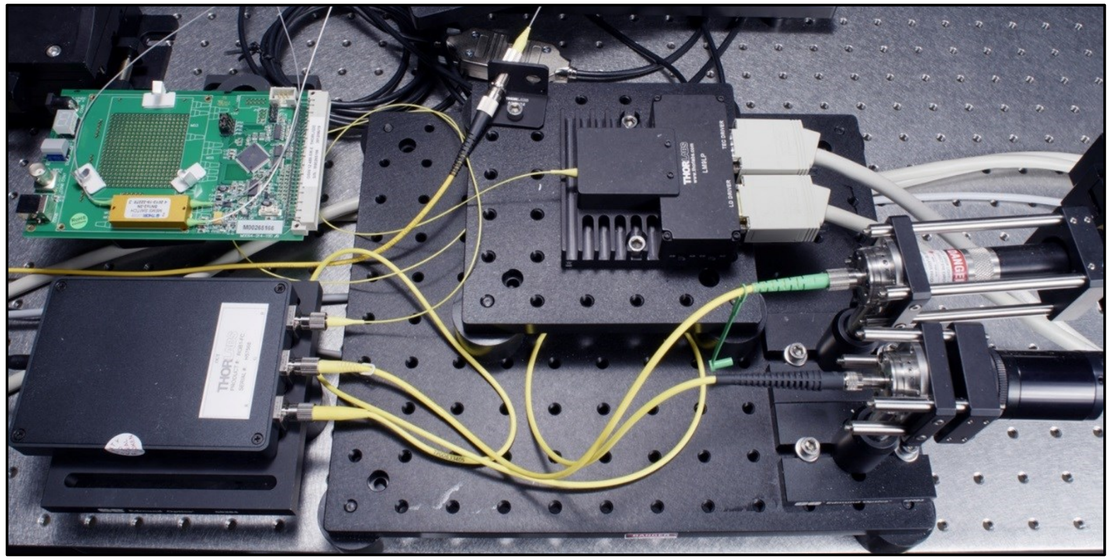
NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis [MildenHall et al. ECCV 2020]

Acquisition of scattering materials

[Gkioulekas et al., 2013]

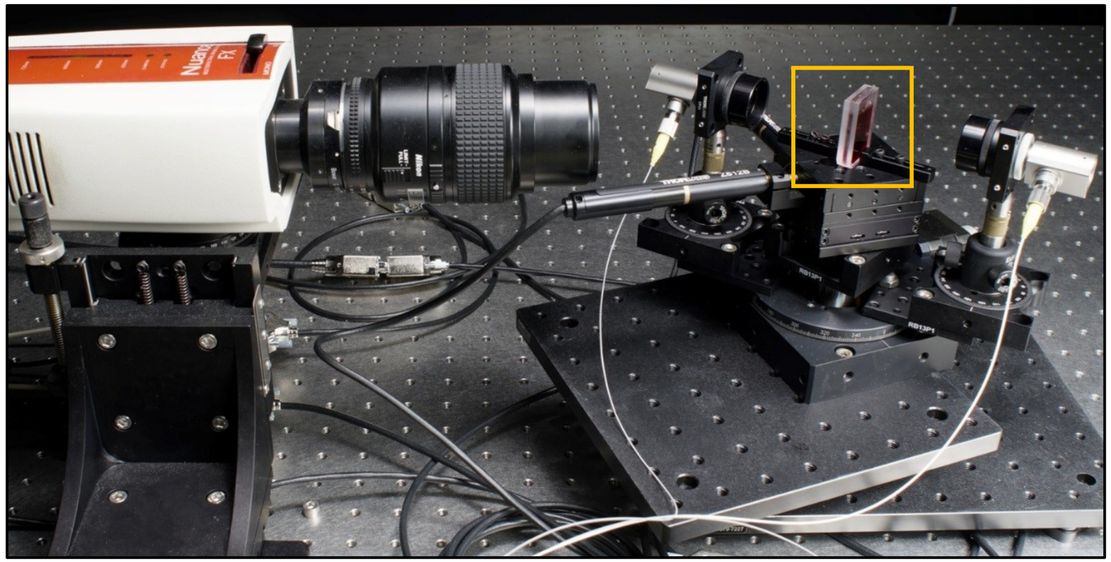
Acquisition setup

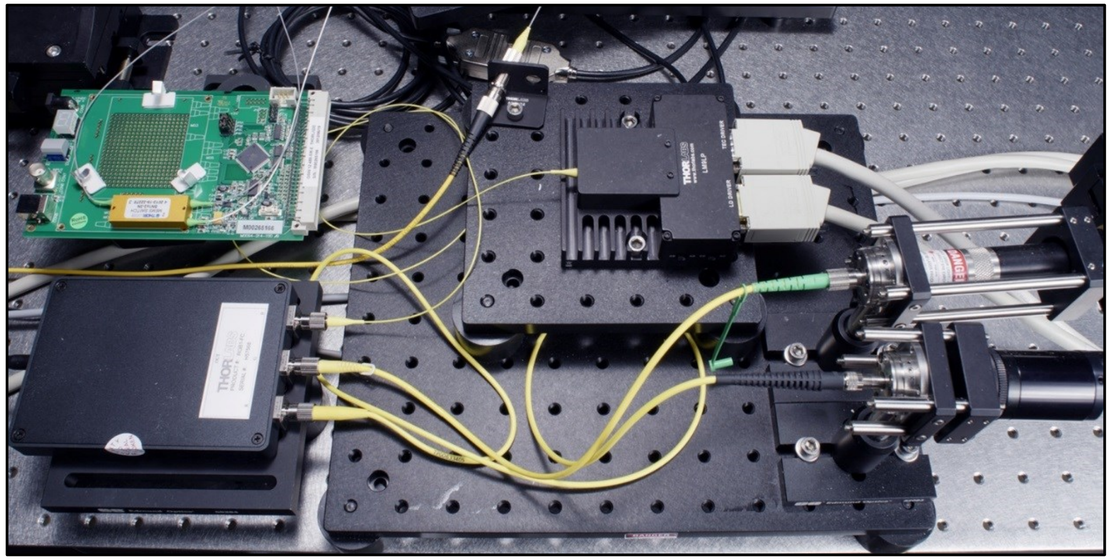


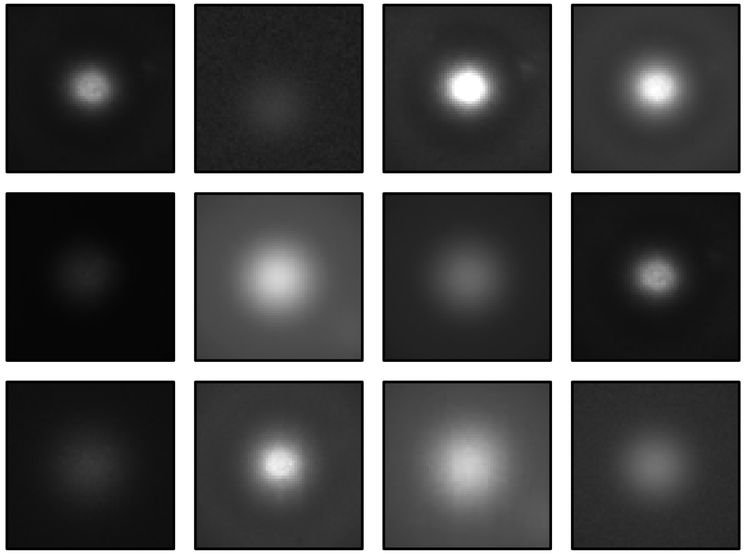


[Gkioulekas et al., 2013]

Acquisition setup

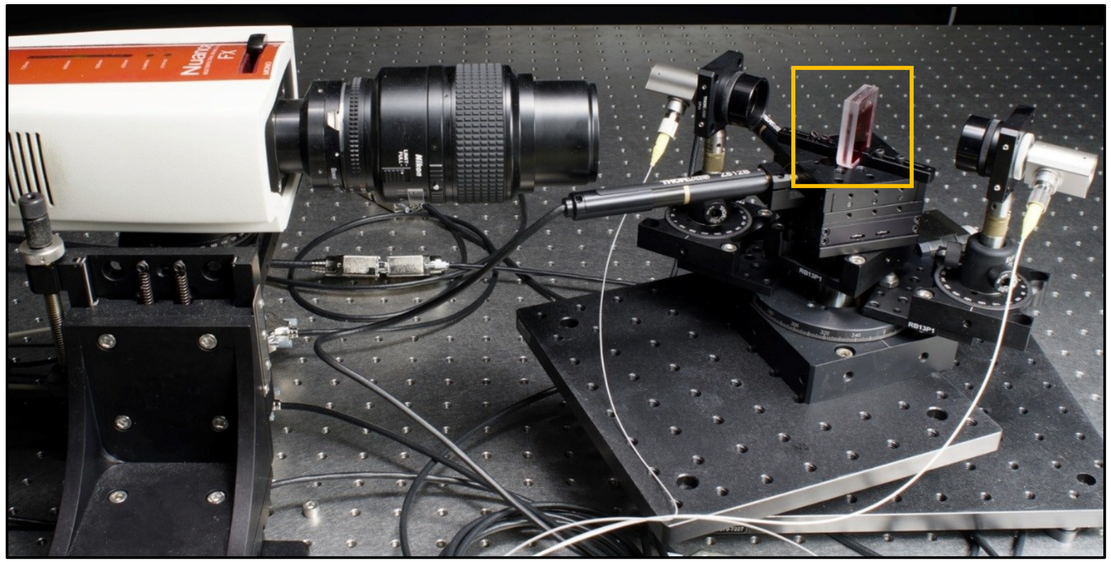


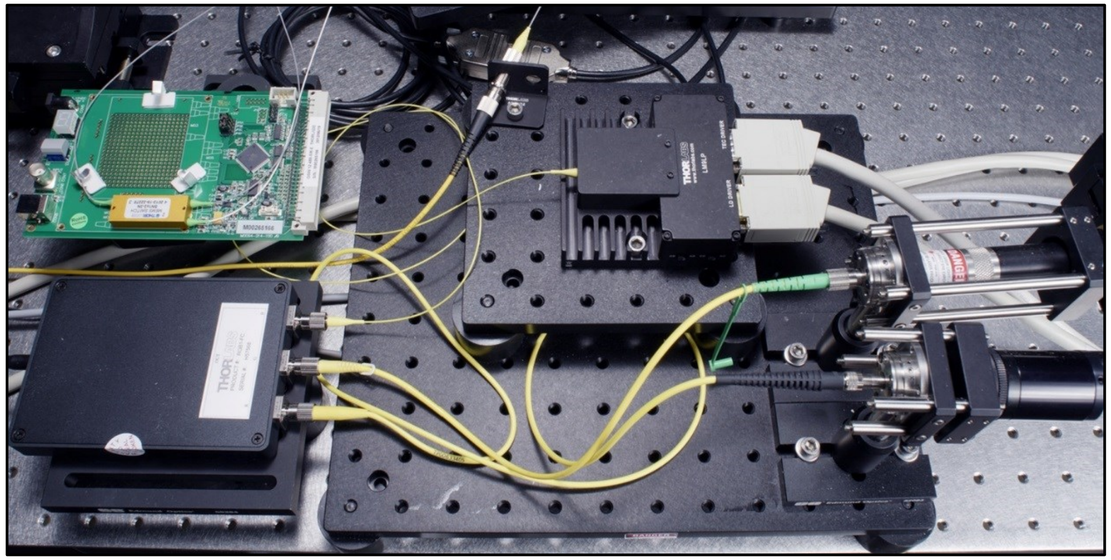


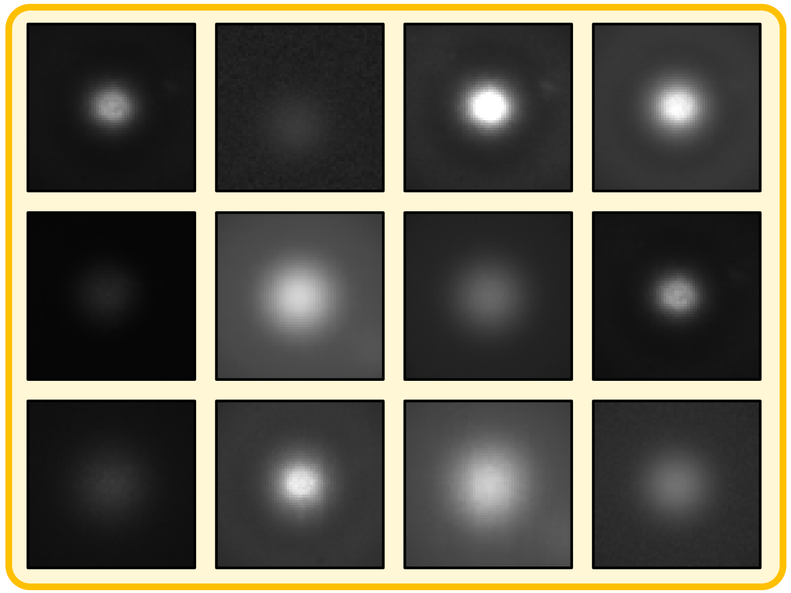


[Gkioulekas et al., 2013]

Acquisition setup



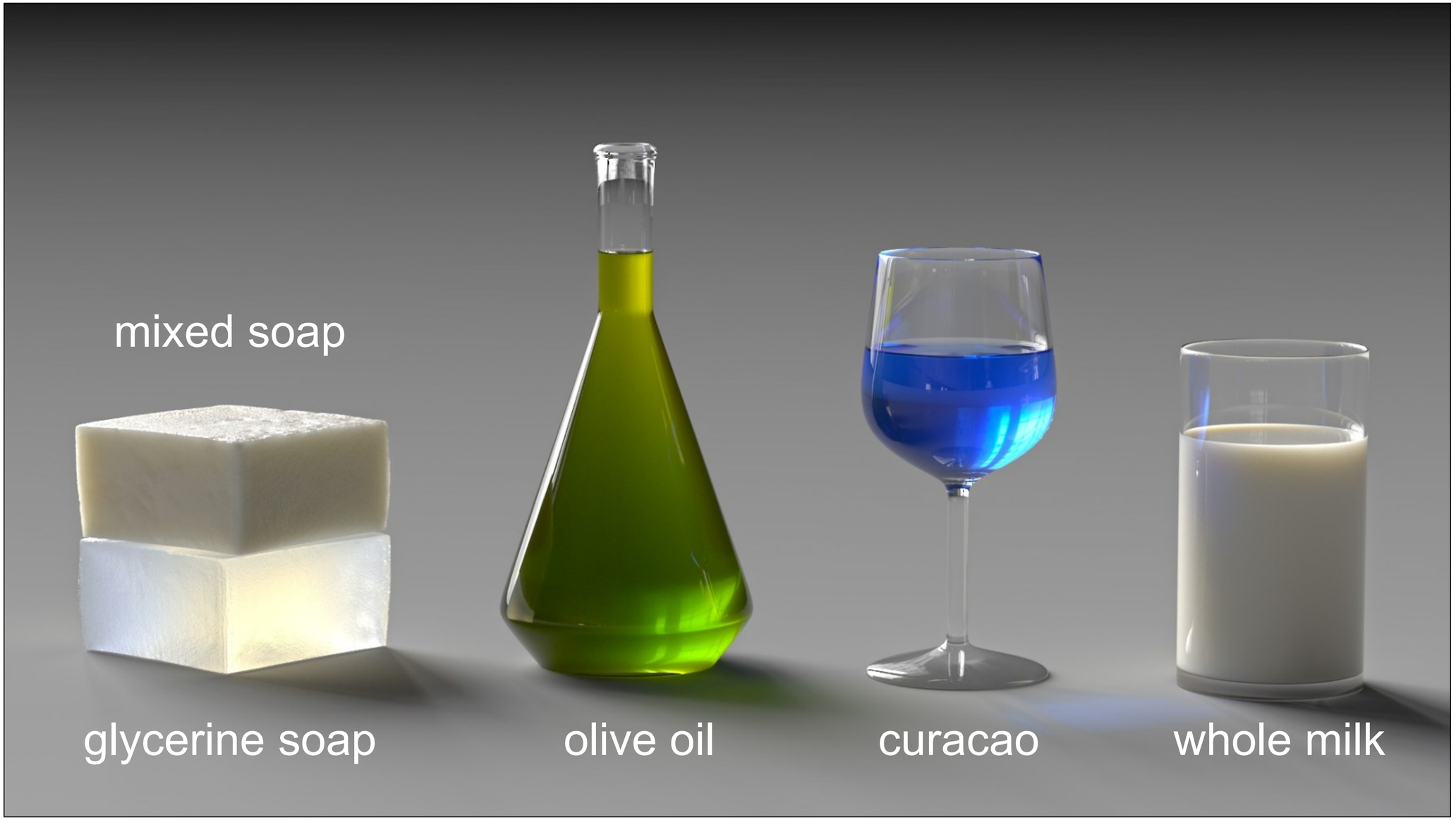




Invert using differentiable rendering

[Gkioulekas et al., 2013]

Synthetic renderings



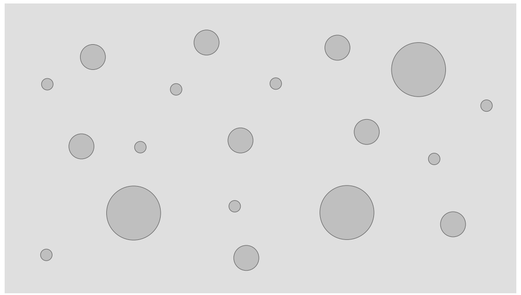
[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

unknown nanodispersion



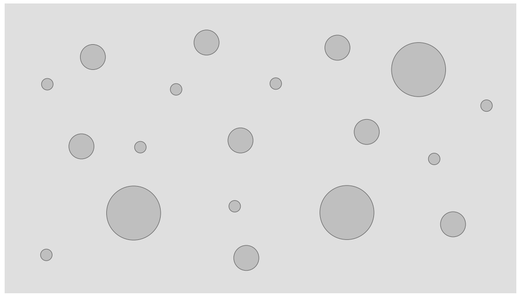
dispersing medium

particle material

[Gkioulekas et al., 2013]

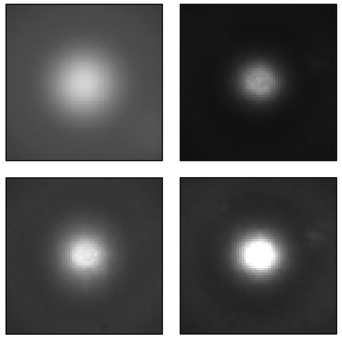
Particle sizing of industrial nanodispersions

unknown nanodispersion

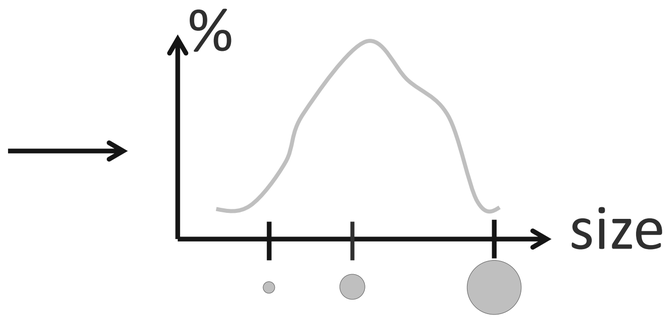


dispersing medium

particle material



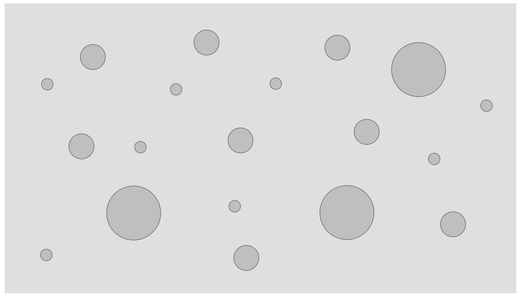
measurements



[Gkioulekas et al., 2013]

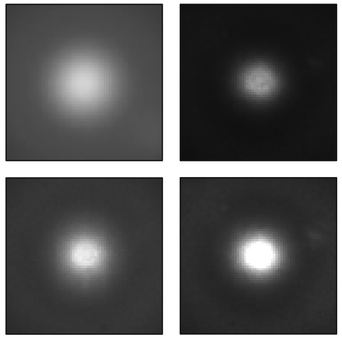
Particle sizing of industrial nanodispersions

unknown nanodispersion

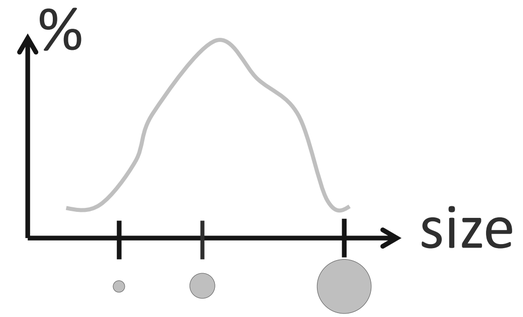


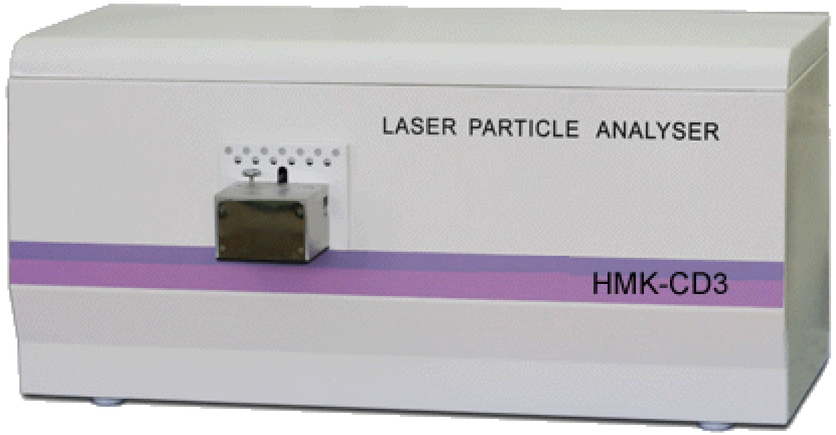
dispersing medium

particle material



measurements





[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

polystyrene

aluminum oxide

very precise dispersions (NIST Traceable Standards)

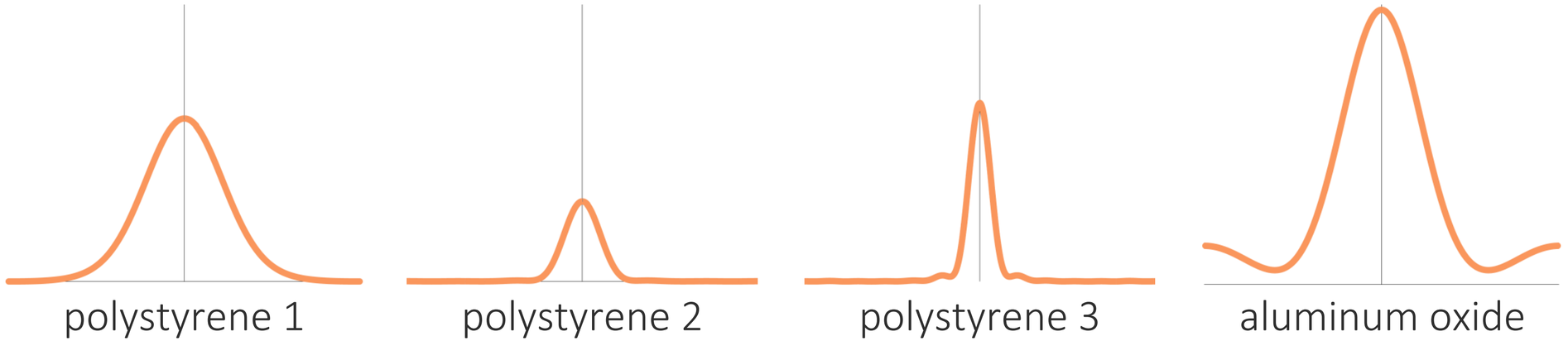
[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

polystyrene

aluminum oxide

very precise dispersions (NIST Traceable Standards)



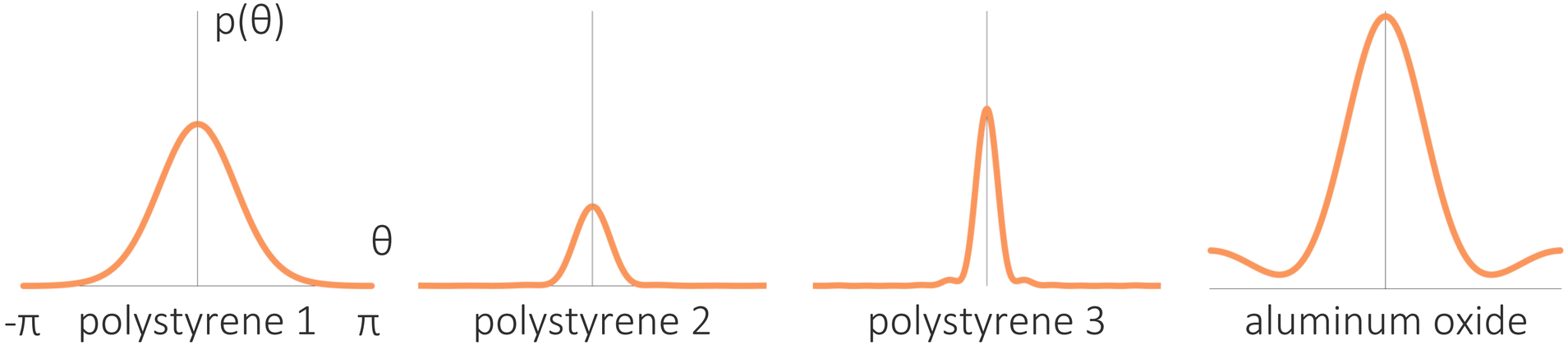
[Gkioulekas et al., 2013]

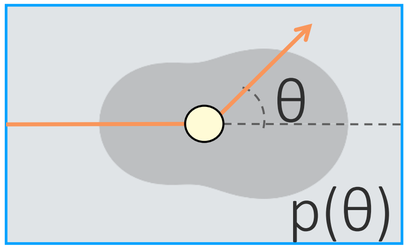
Particle sizing of industrial nanodispersions

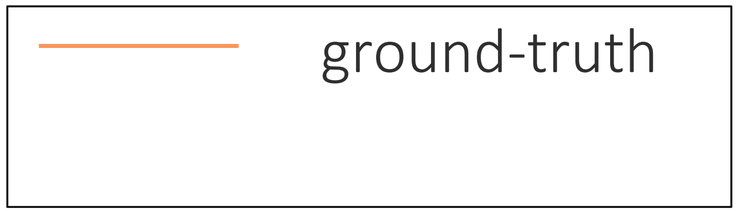
polystyrene

aluminum oxide

very precise dispersions (NIST Traceable Standards)







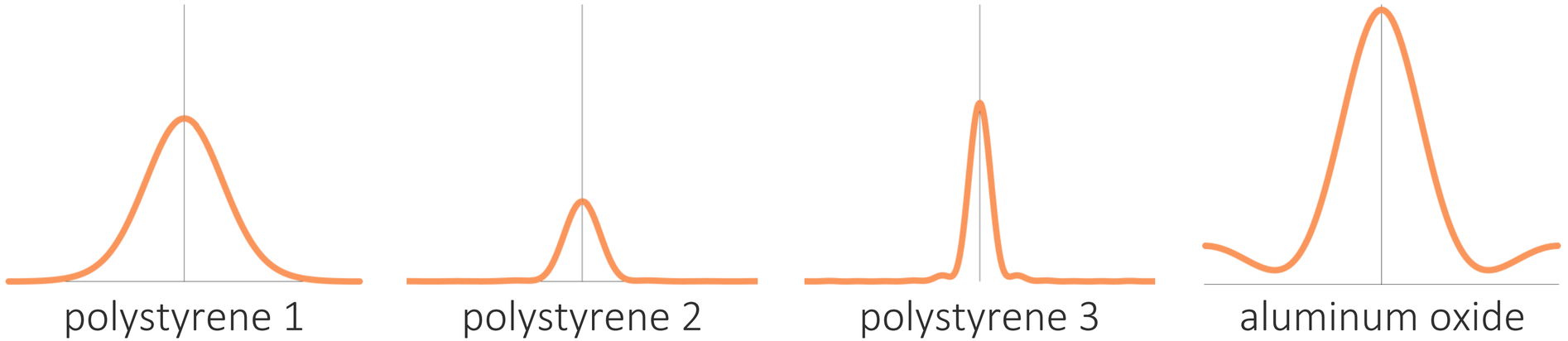
[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

polystyrene

aluminum oxide

very precise dispersions (NIST Traceable Standards)



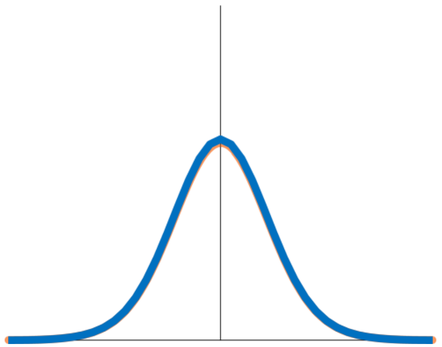
[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

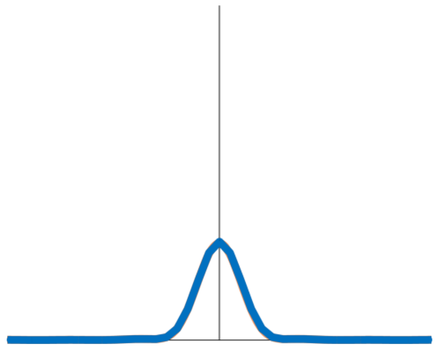
polystyrene

aluminum oxide

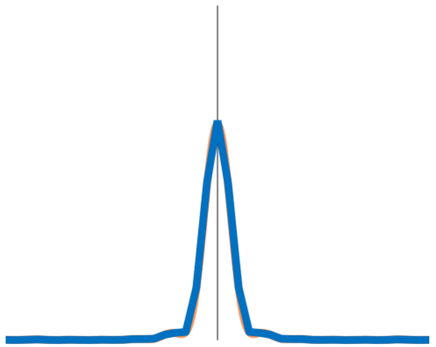
very precise dispersions (NIST Traceable Standards)



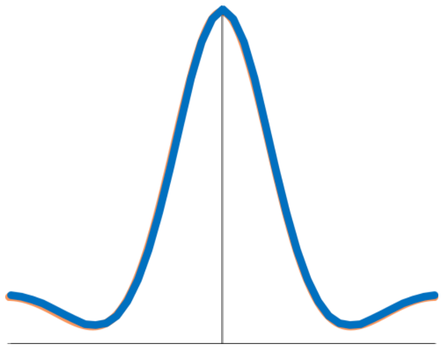
polystyrene 1



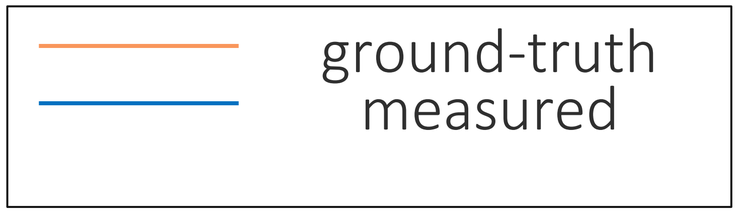
polystyrene 2



polystyrene 3



aluminum oxide



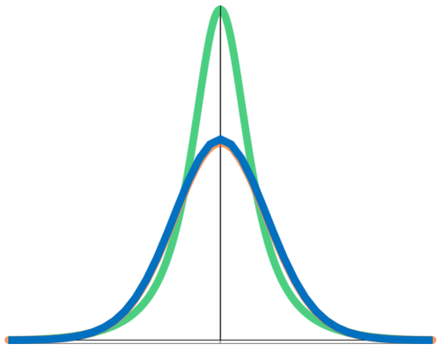
[Gkioulekas et al., 2013]

Particle sizing of industrial nanodispersions

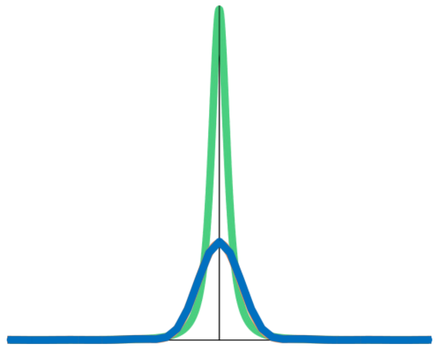
polystyrene

aluminum oxide

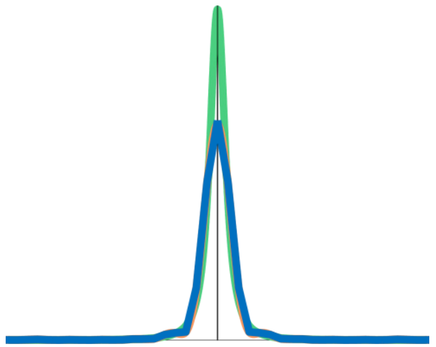
very precise dispersions (NIST Traceable Standards)



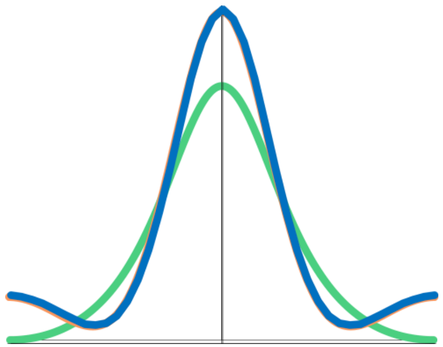
polystyrene 1



polystyrene 2



polystyrene 3

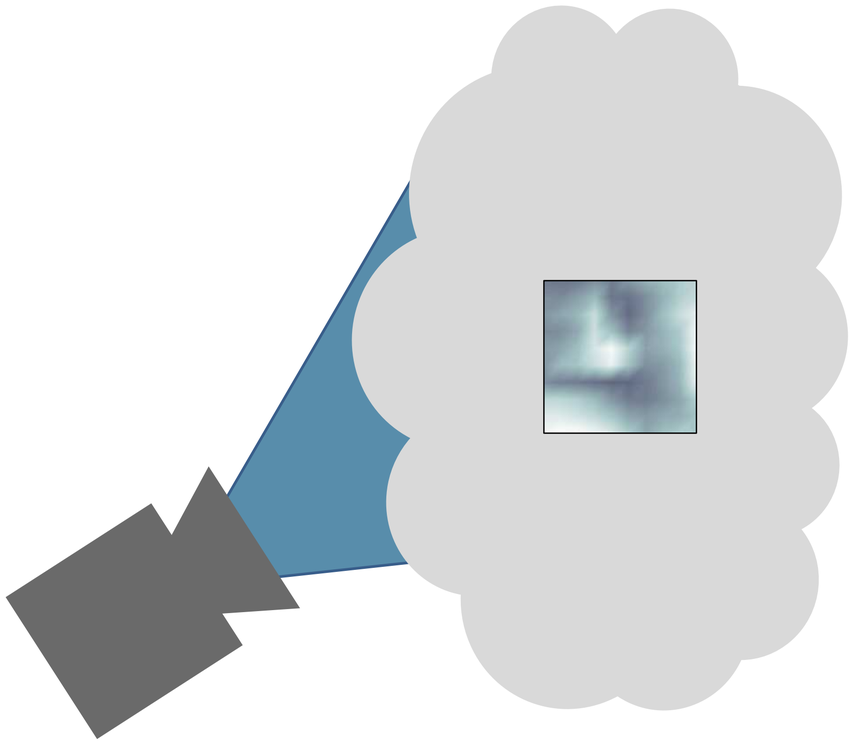


aluminum oxide

ground-truth
measured
previous fit

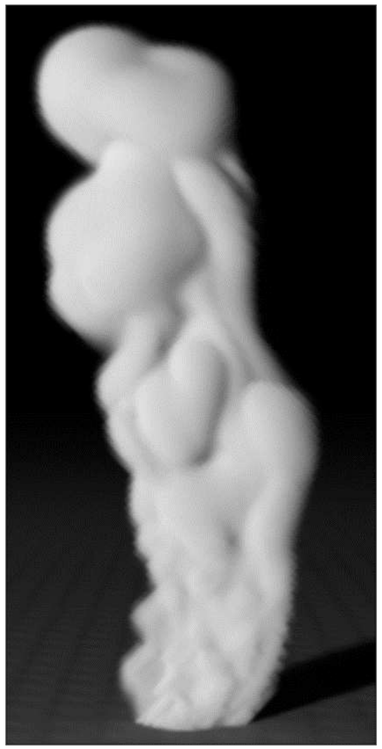
[Gkioulekas et al., 2013]

Optical tomography



camera

thick smoke cloud



simulated camera measurements

reconstructed cloud volume

slice through the cloud

[Gkioulekas et al., 2015]

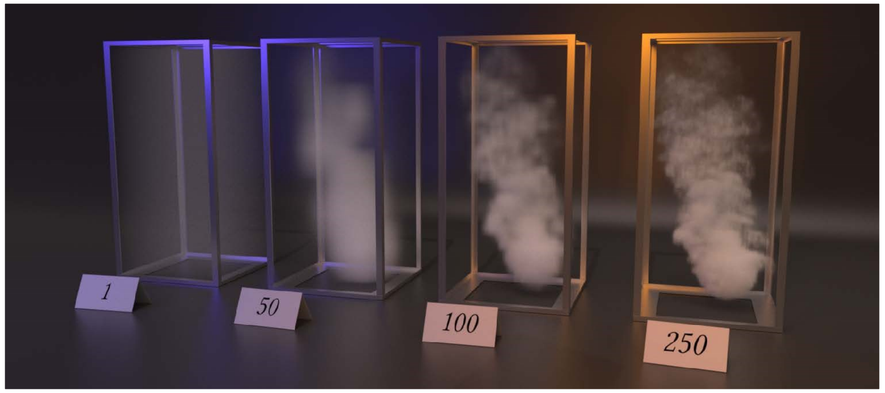
Computational Fabrication

Determining the material configuration for individual voxels in full-color inkjet 3D printing

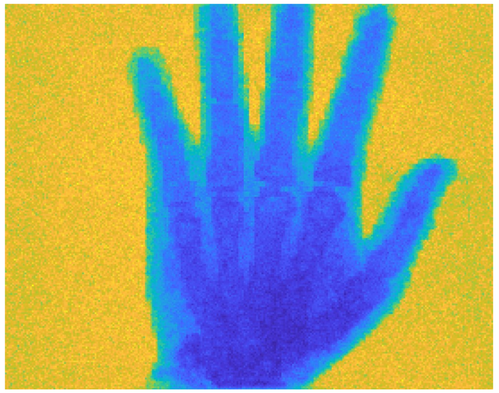
[Nindel et al. 2021]

Active area of research

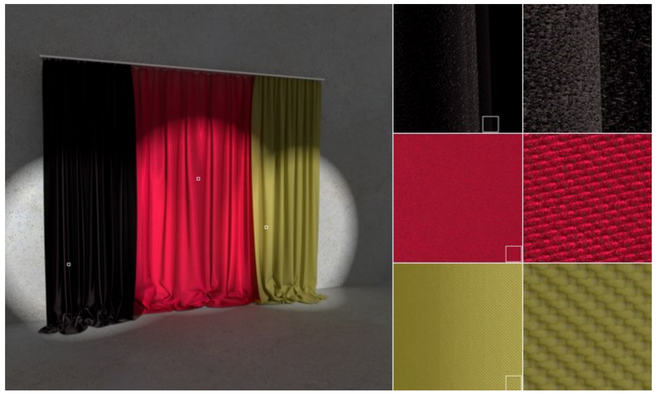
industrial dispersions
[Gkioulekas et al. 2013]



efficient algorithms
[Nimier-David et al. 2019, 2020]

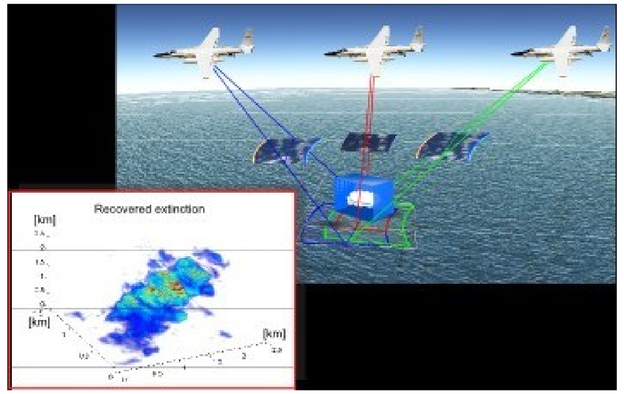


computed tomography
[Geva et al. 2018]



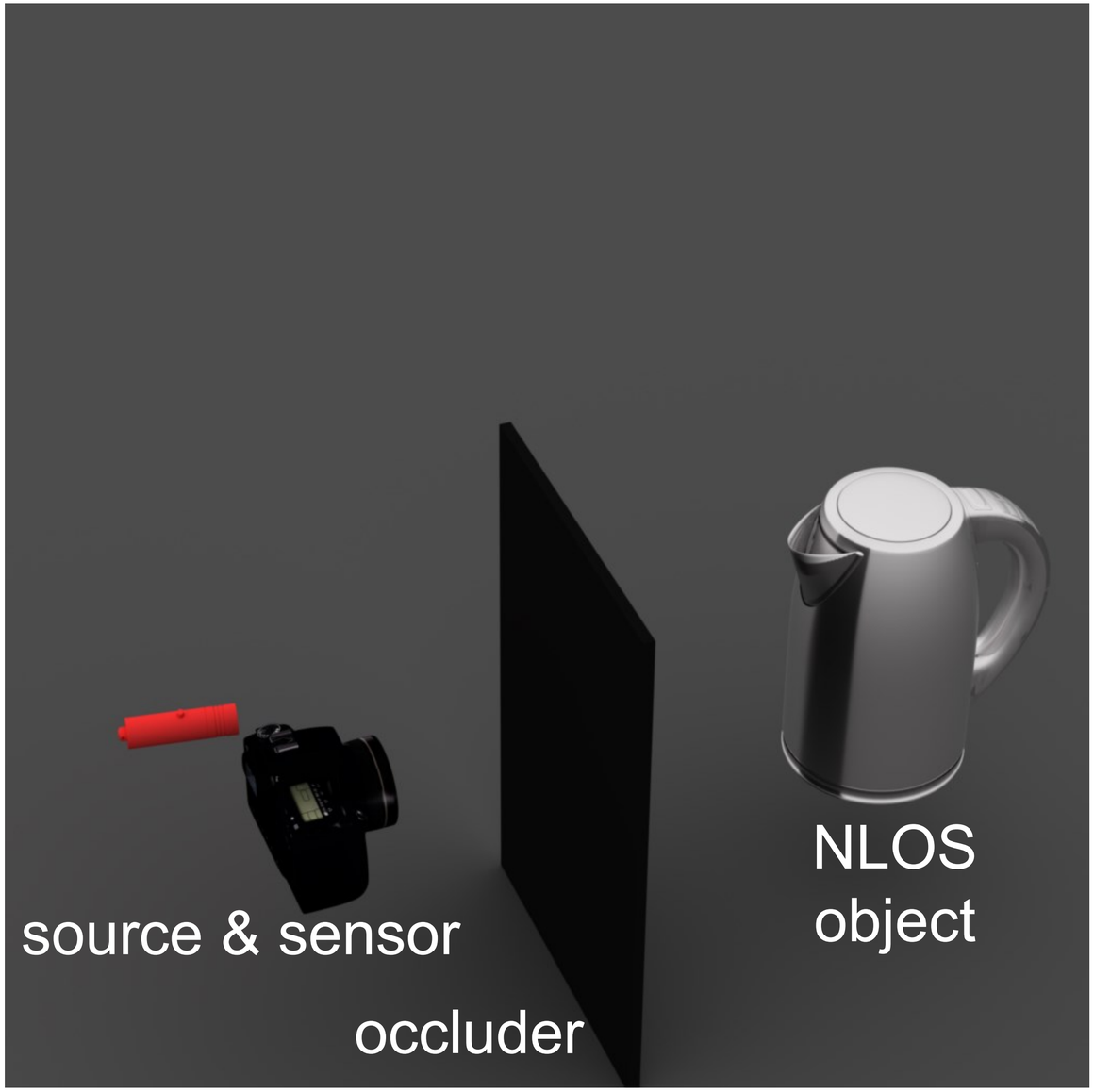
woven fabrics
[Khungurn et al. 2015,
Zhao et al. 2016]

3D printing
[Elek et al. 2019,
Nindel et al. 2021]

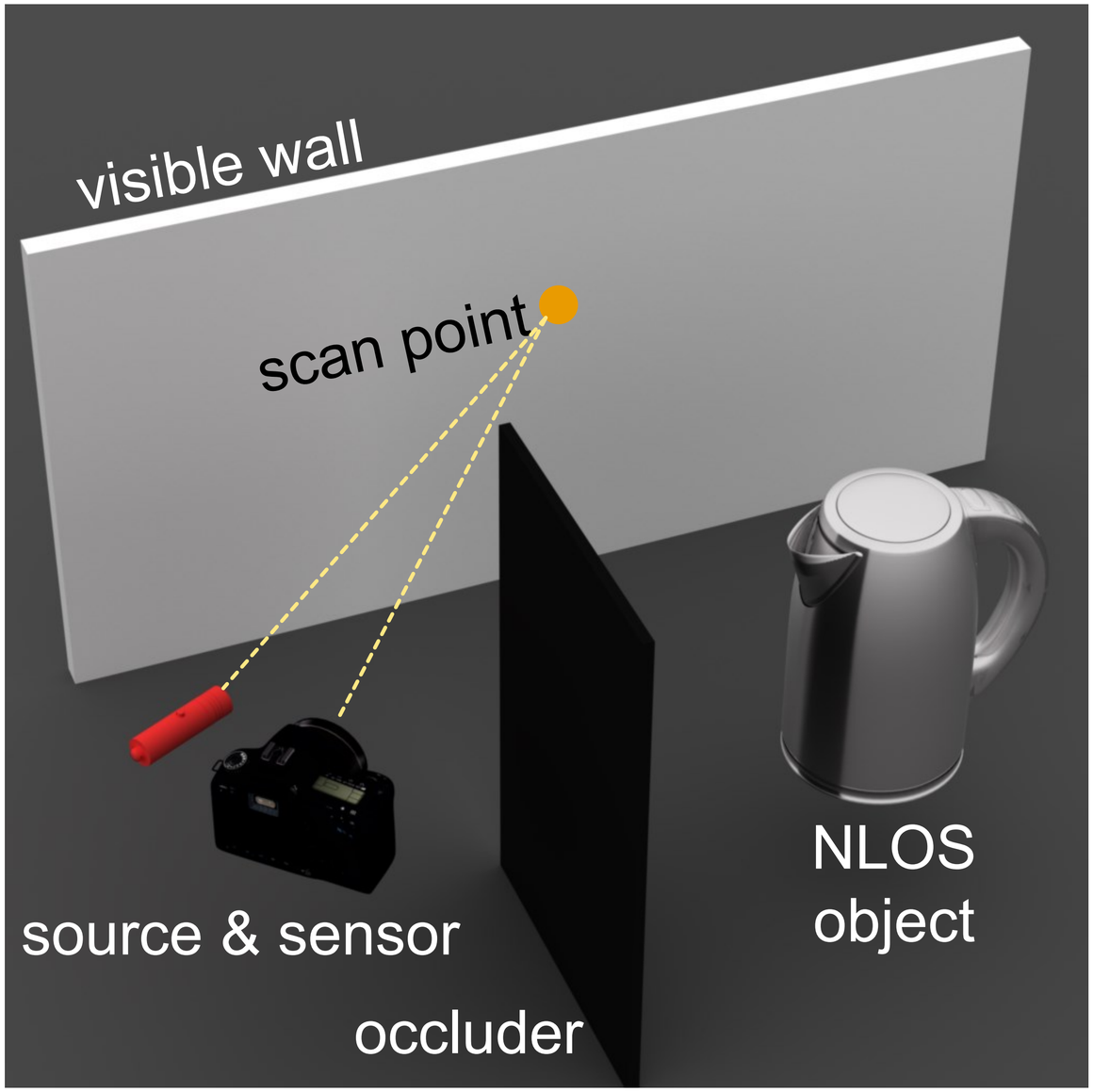


cloud tomography
[Levis et al. 2015,
2017, 2020]

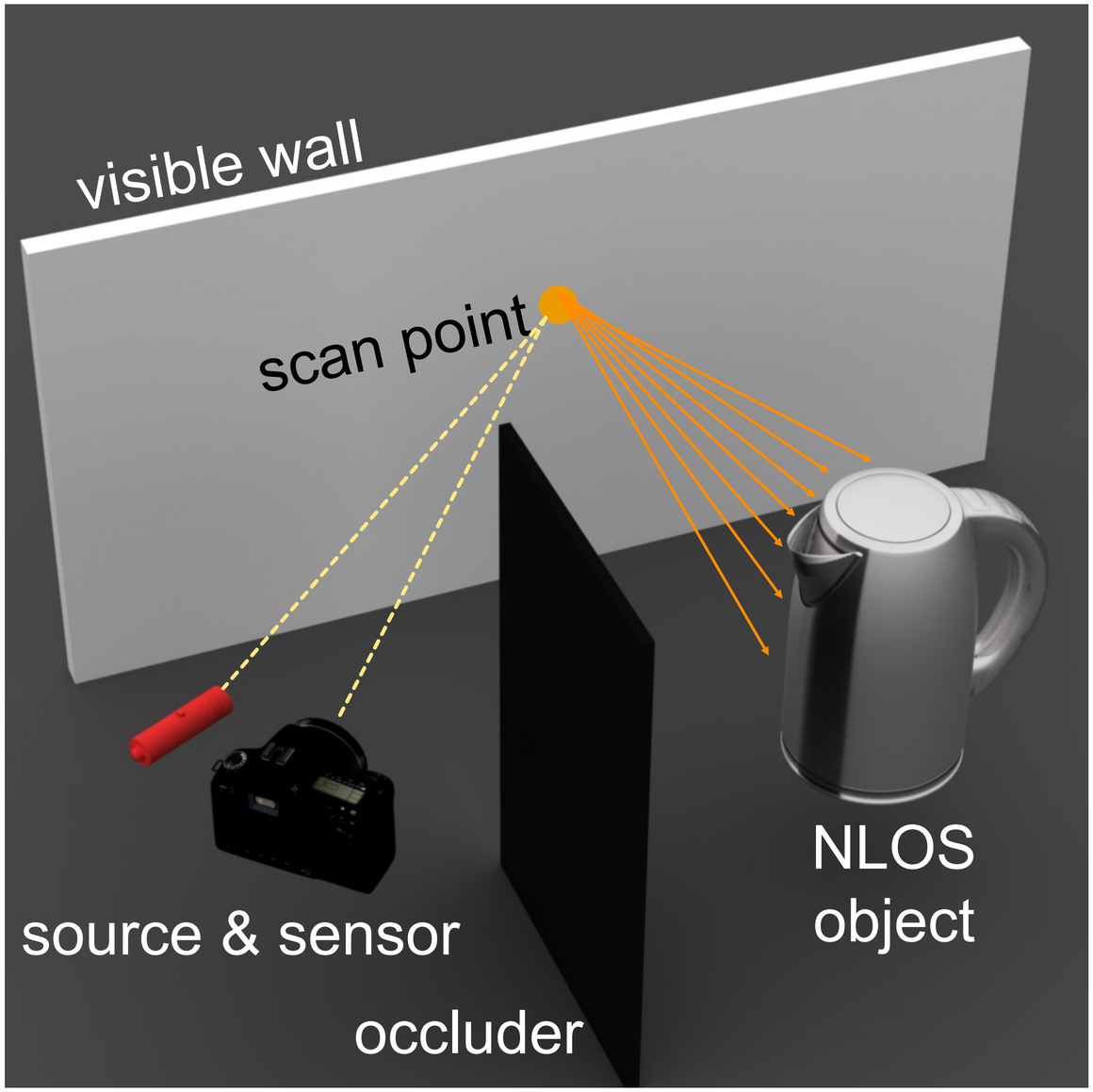
Non-line-of-sight (NLOS) imaging



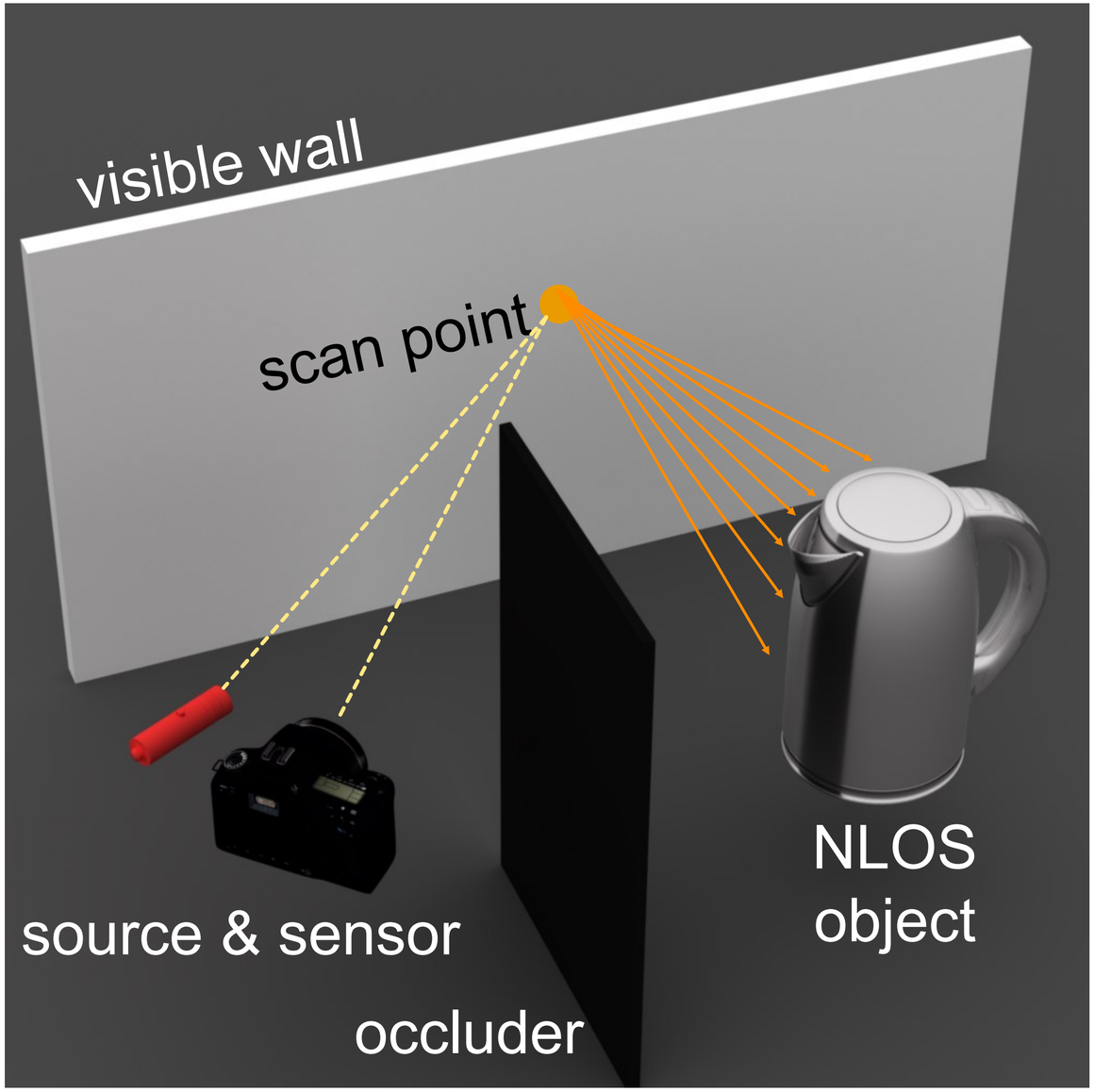
Non-line-of-sight (NLOS) imaging

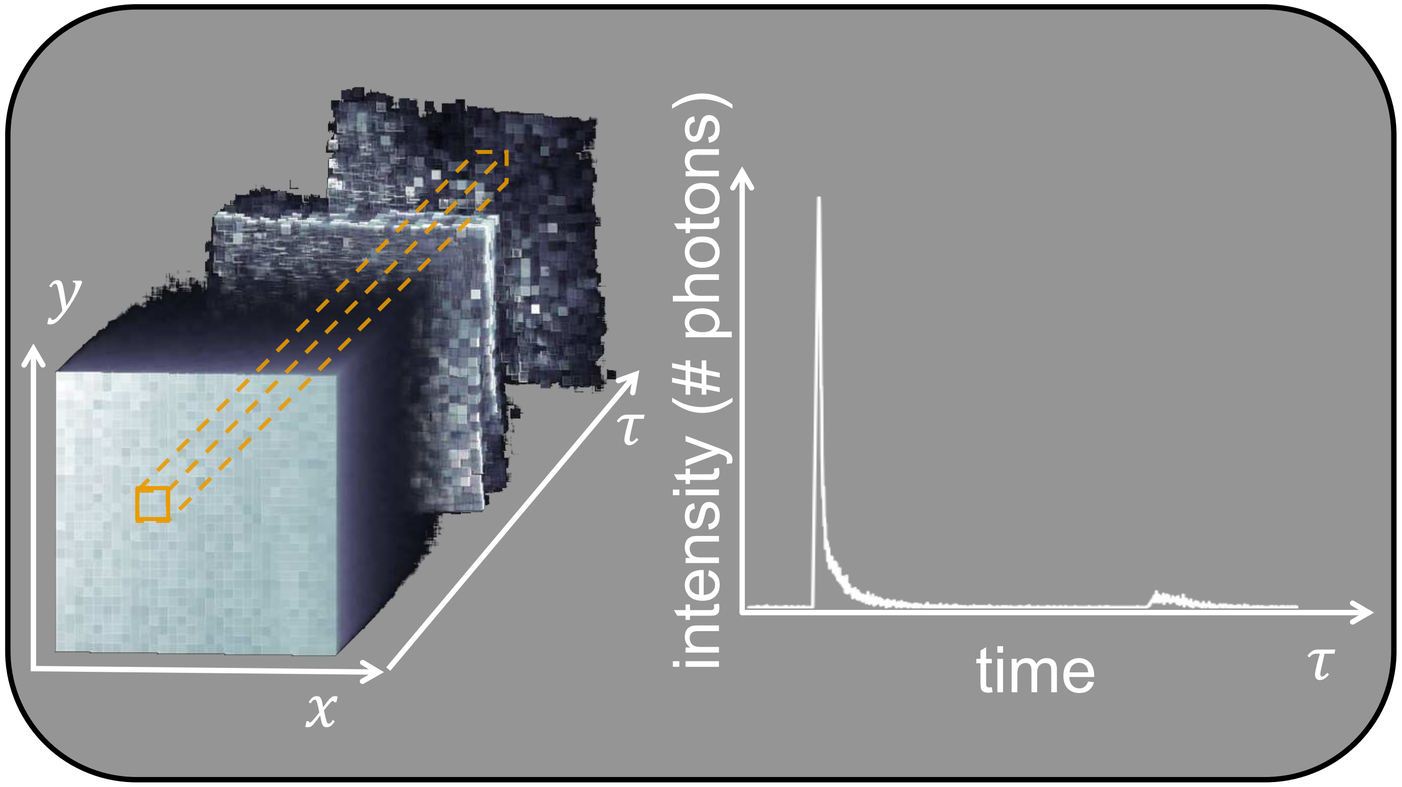


Non-line-of-sight (NLOS) imaging



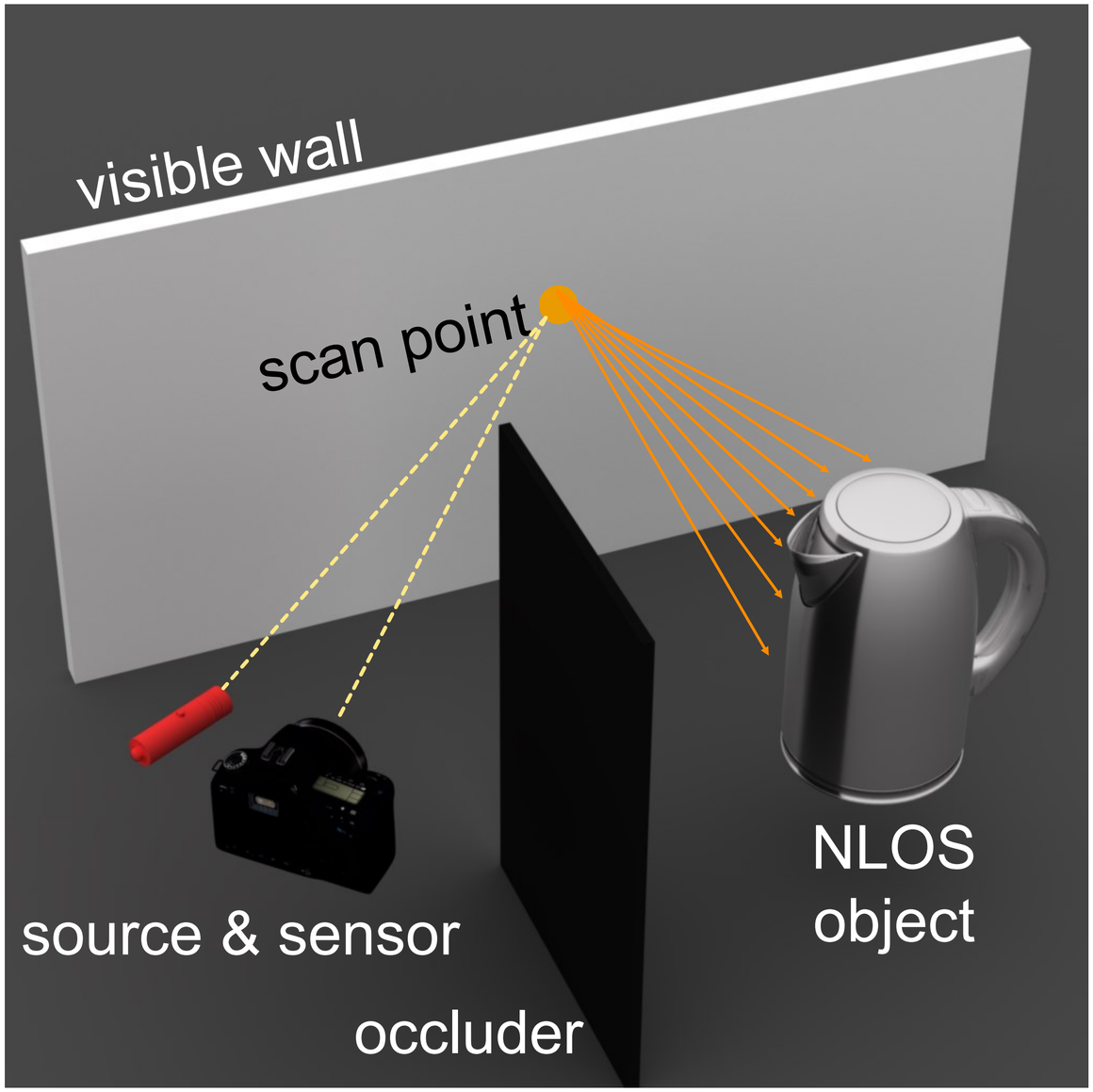
Non-line-of-sight (NLOS) imaging

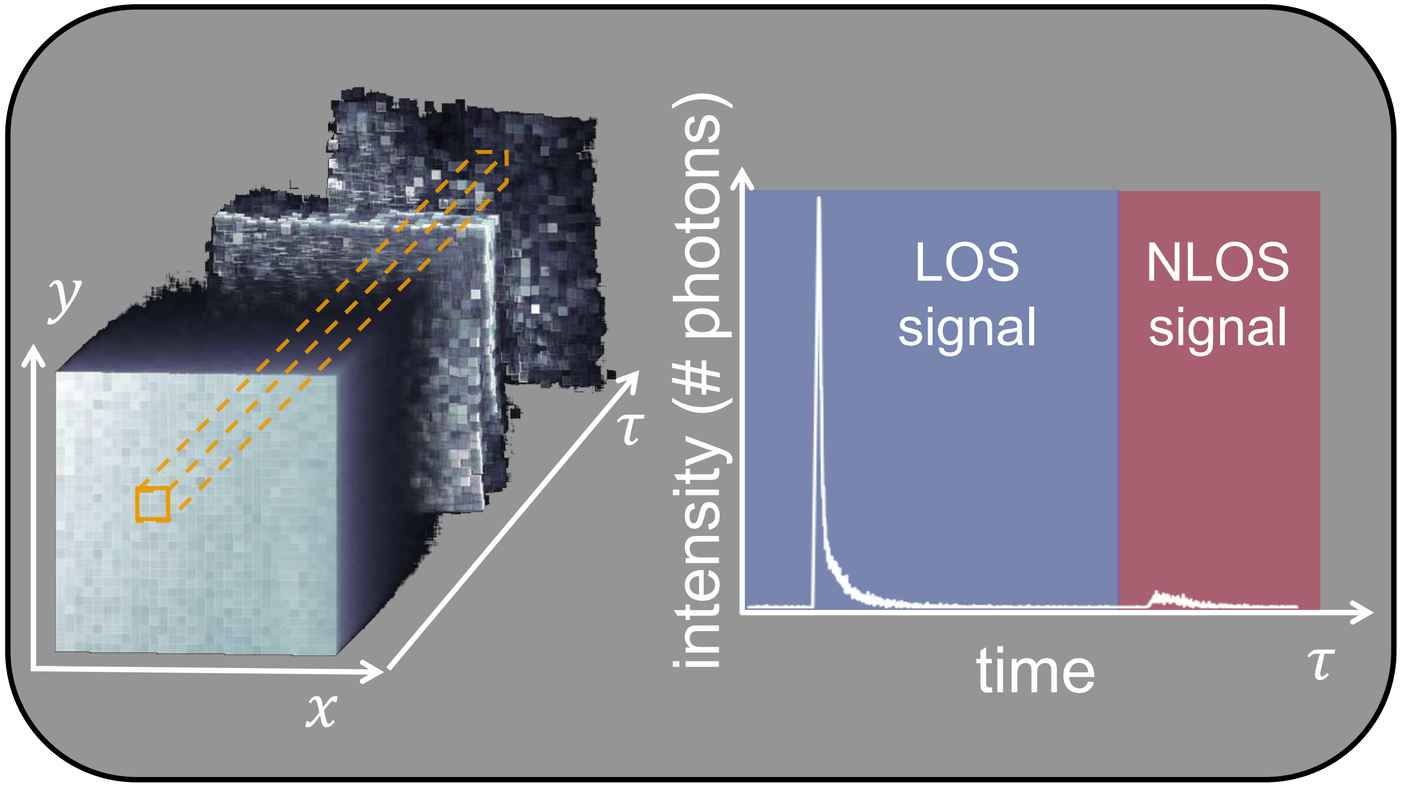




Time-of-flight measurements

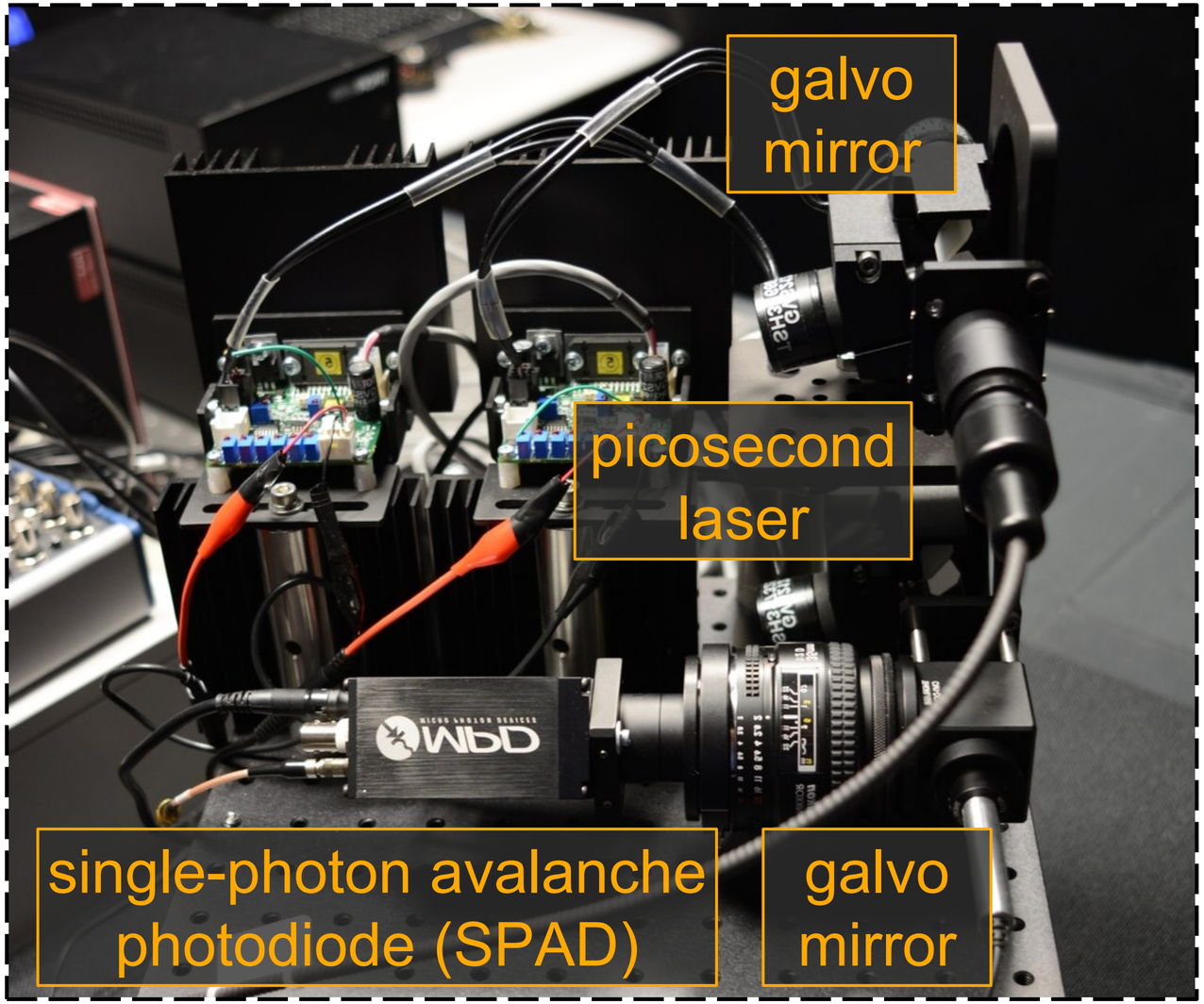
Non-line-of-sight (NLOS) imaging

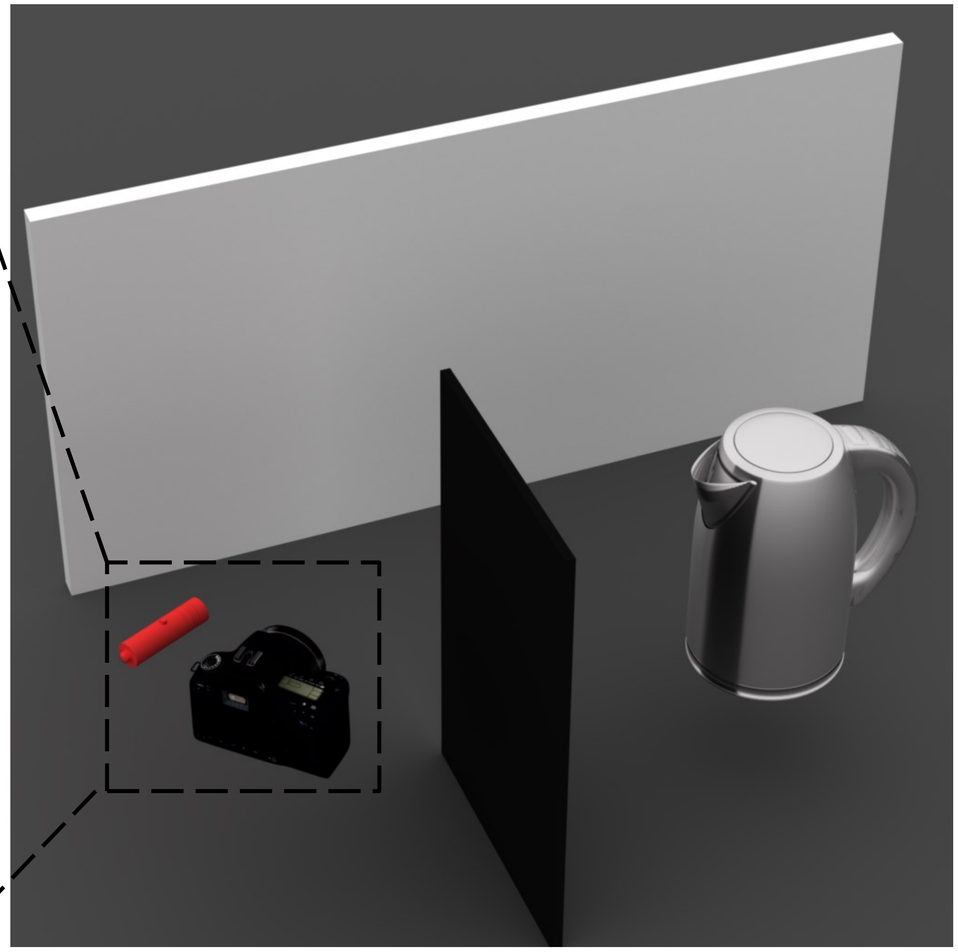




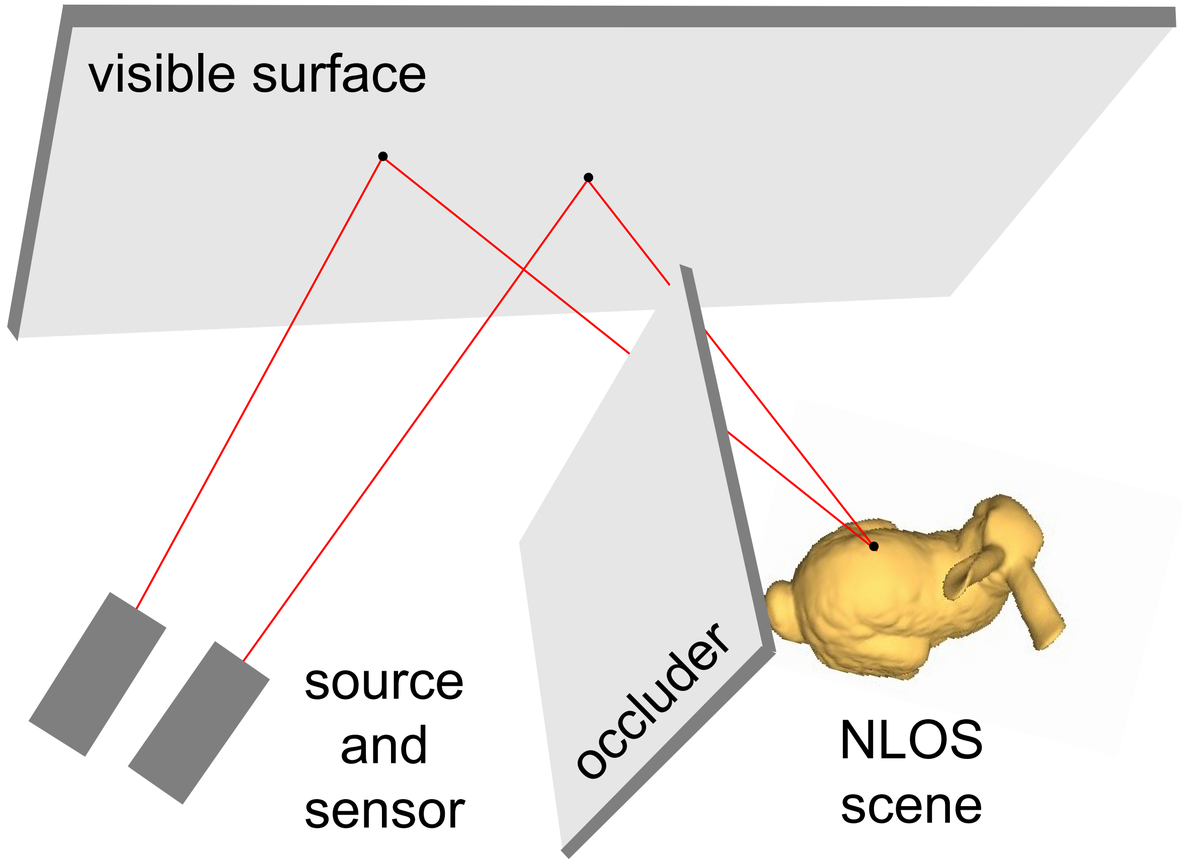
Time-of-flight measurements

SPAD-based lidar

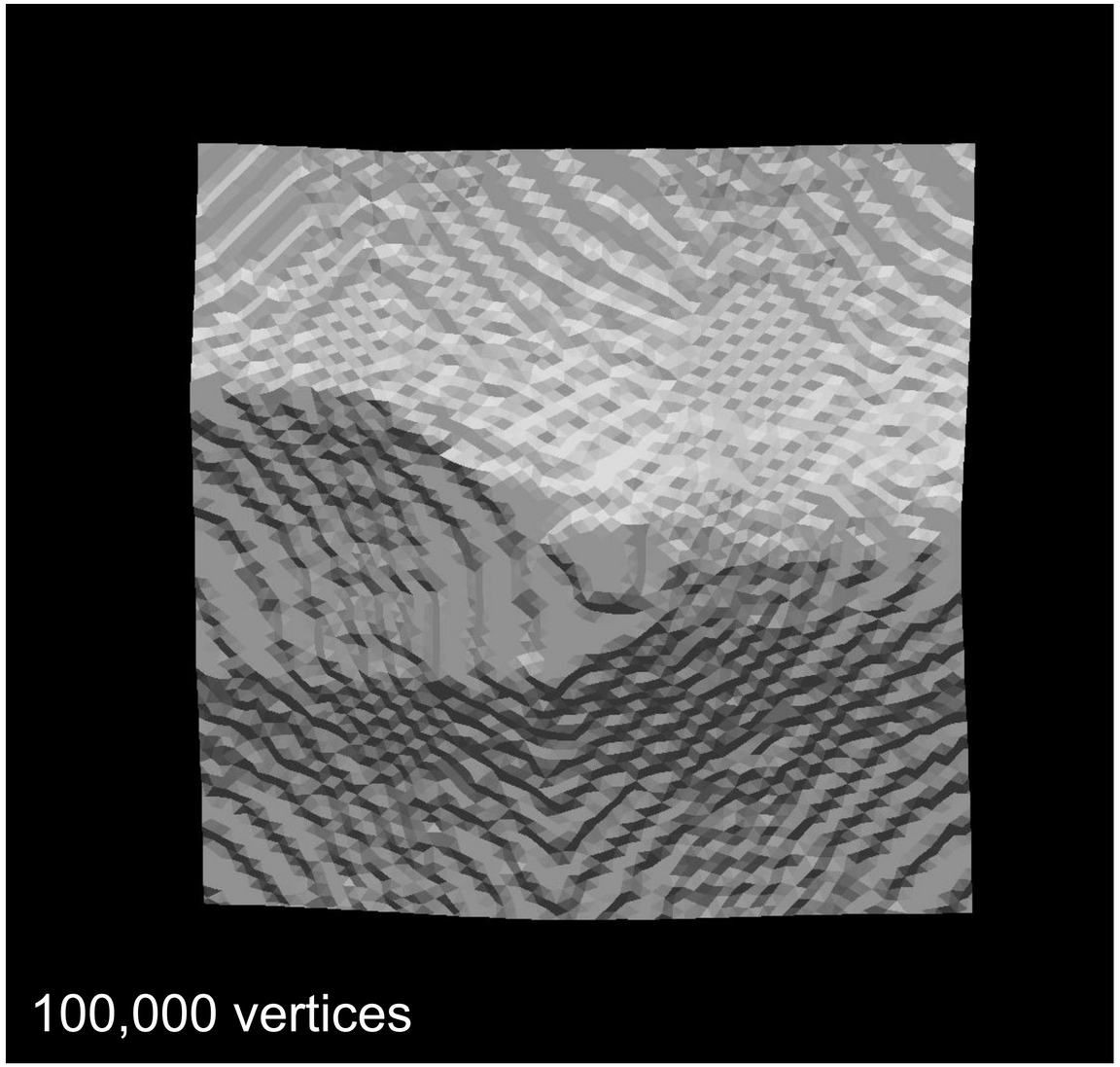




NLOS shape optimization

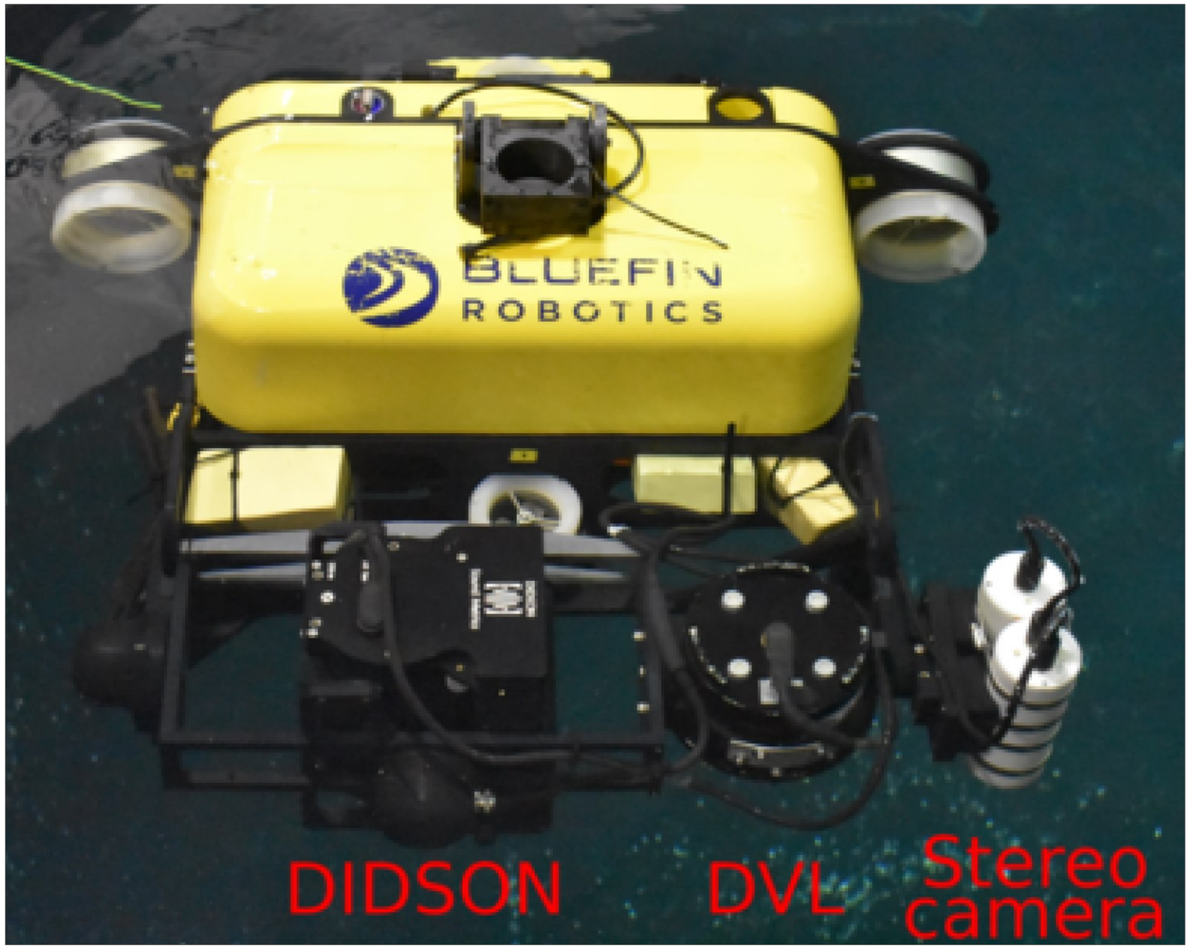


Simulated time-of-flight data

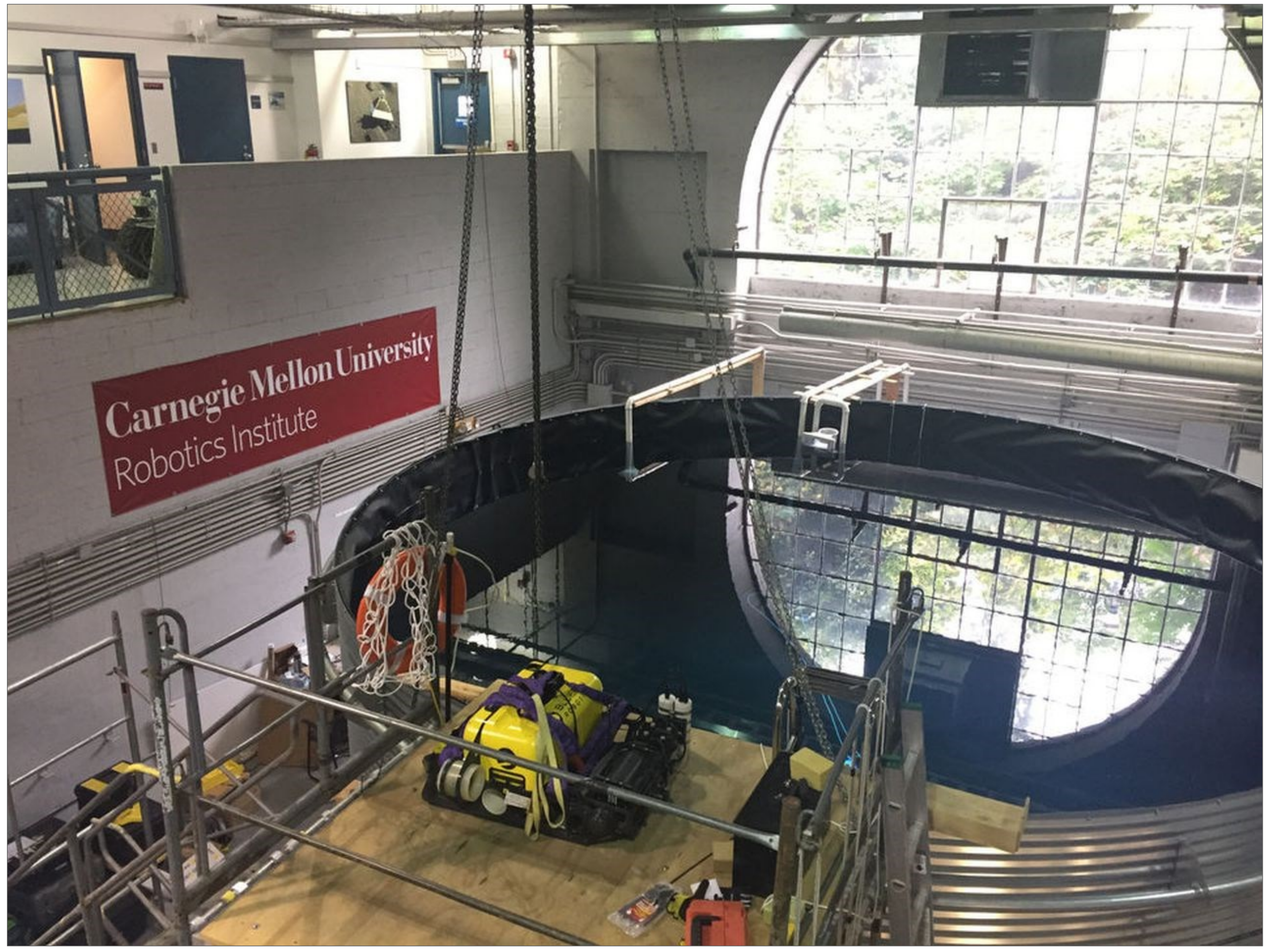


[Tsai et al., 2019]

Underwater 3D using imaging SONAR



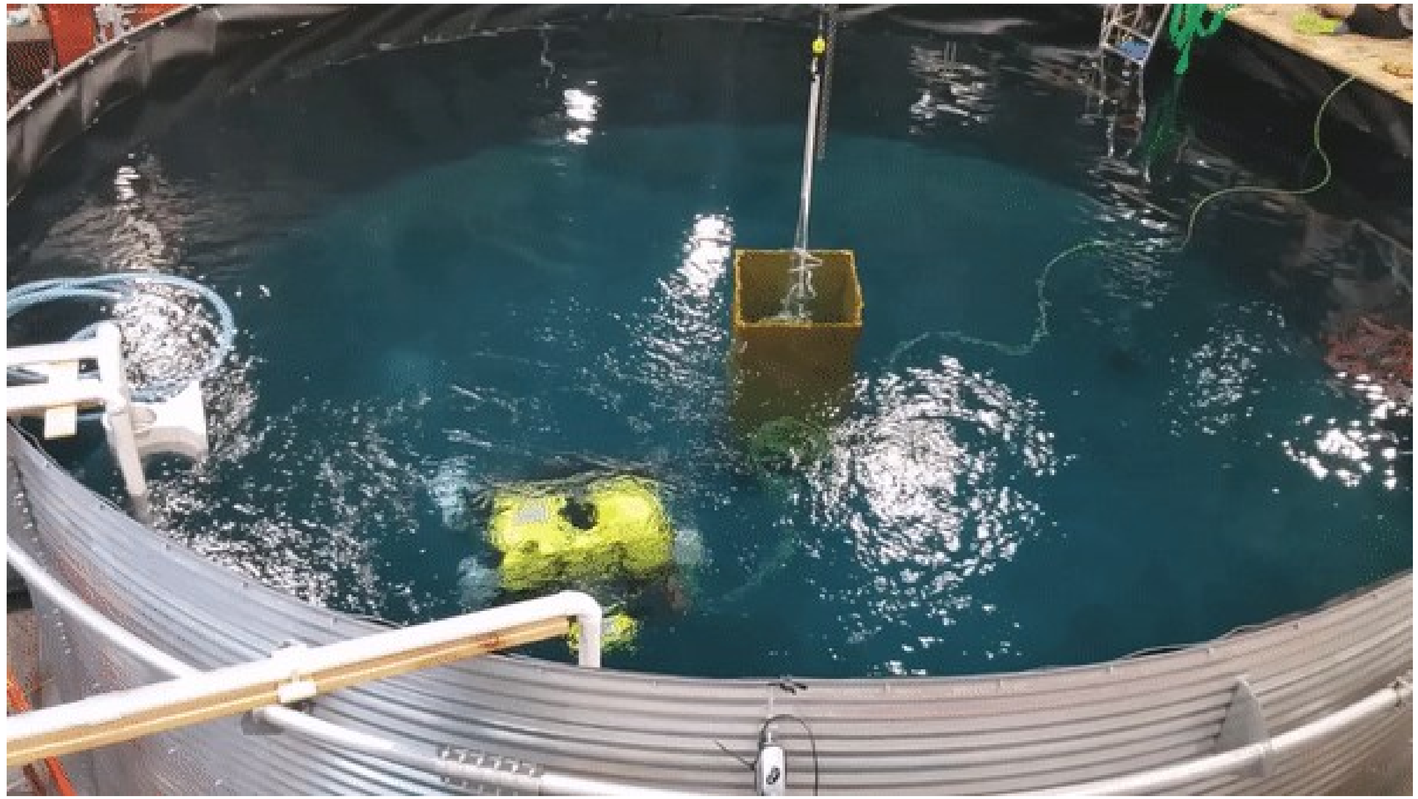
Underwater robot with sonar



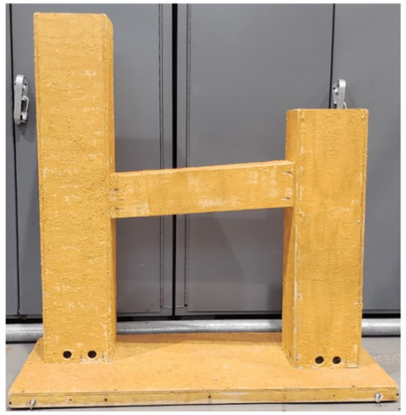
Robotics Institute High Bay

[Qadri et al., 2023]

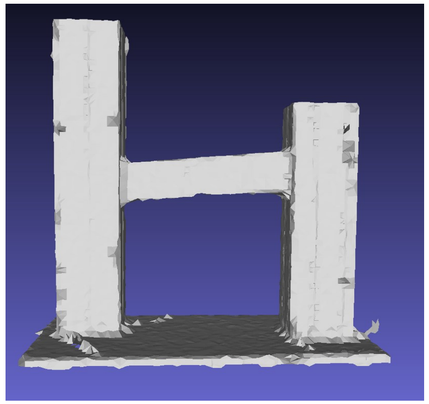
Underwater 3D using imaging SONAR



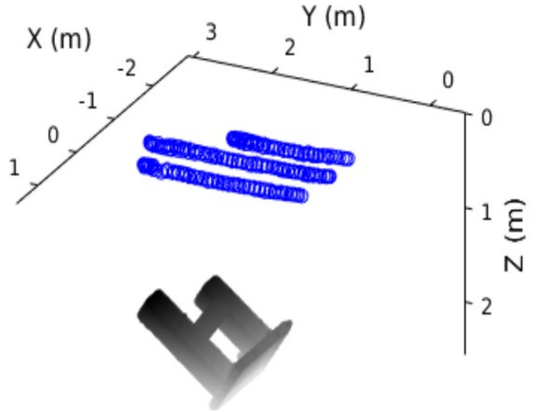
Test structure



Ground truth mesh obtained using a laser scan

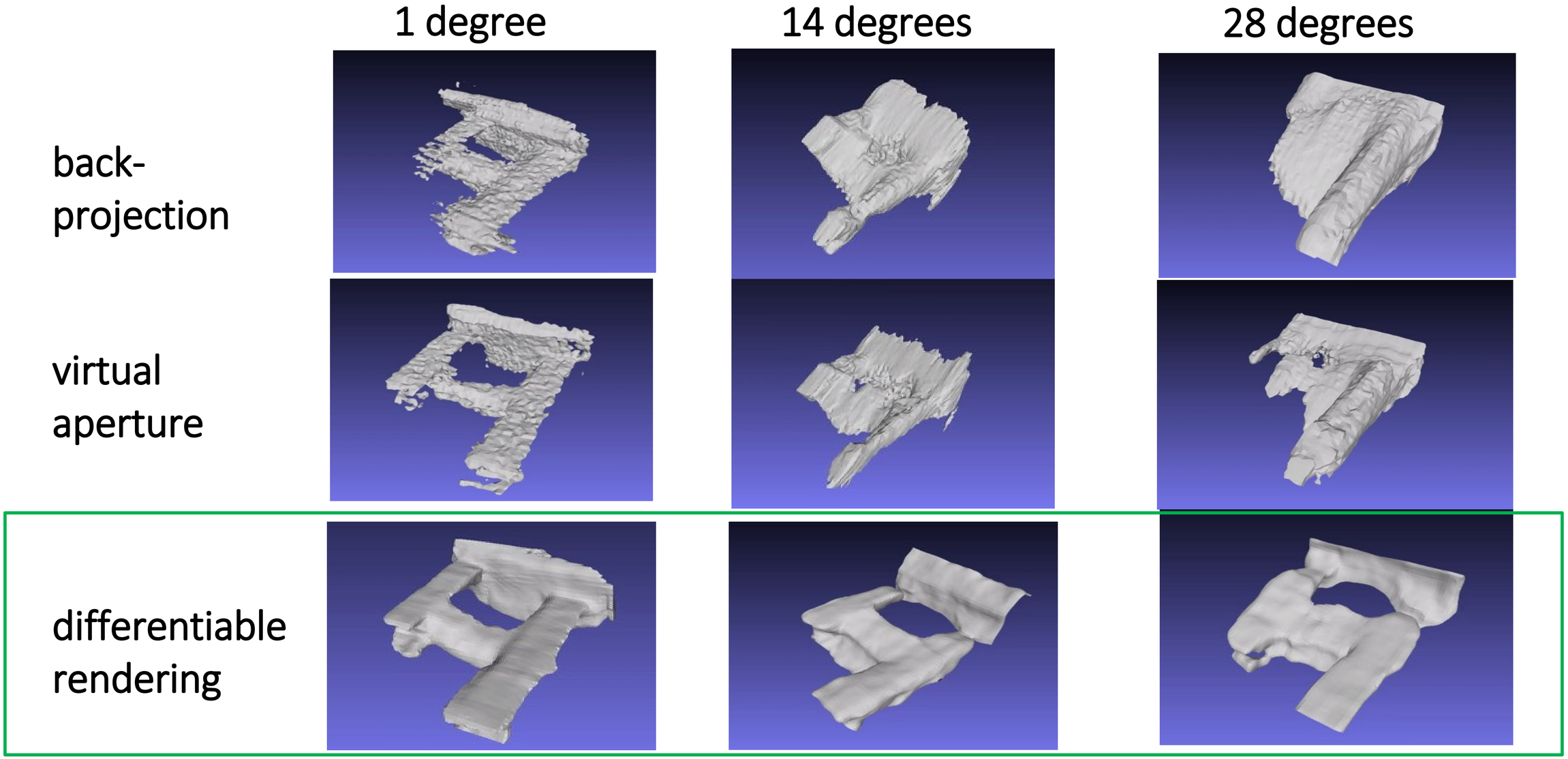


Sonar image collection points



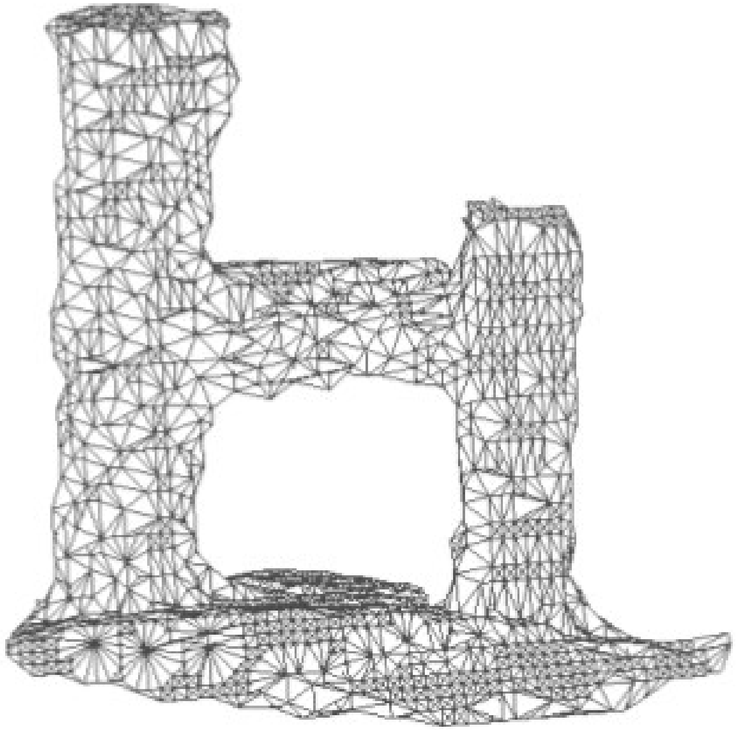
[Qadri et al., 2023]

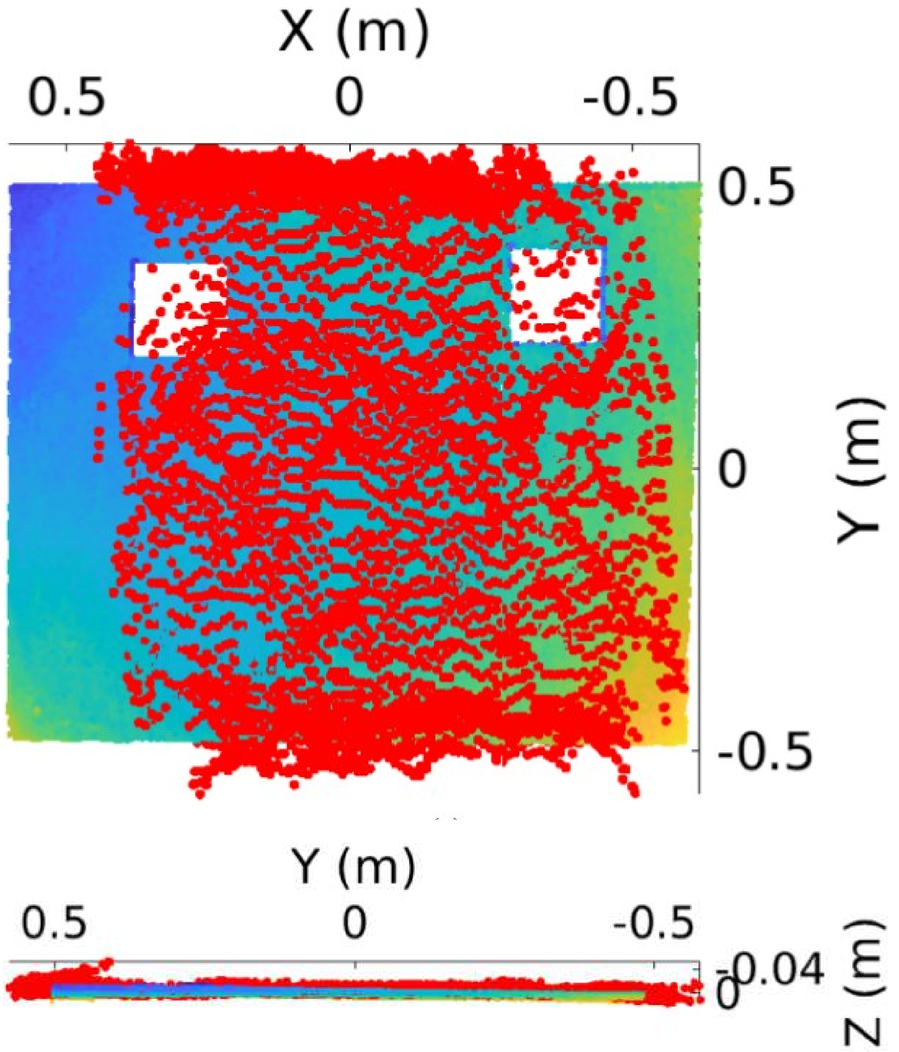
Underwater 3D using imaging SONAR



[Qadri et al., 2023]

Underwater 3D using imaging SONAR





Millimeter-accuracy underwater 3D reconstructions using data captured with an acoustic sonar mounted on robot.

[Qadri et al., 2023]

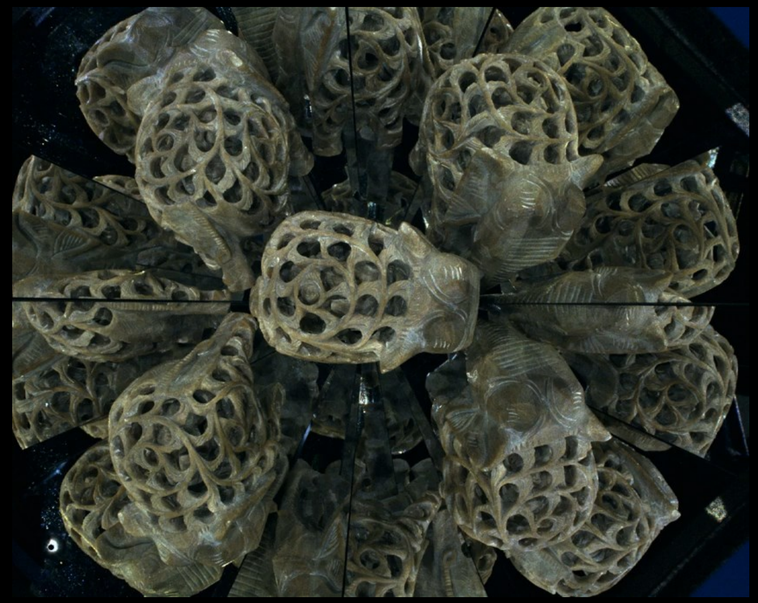
Kaleidoscopic 3D scanning

kaleidoscopic system

[Ahn et al., CVPR 2023, Tuesday PM]

Kaleidoscopic 3D scanning

kaleidoscopic system

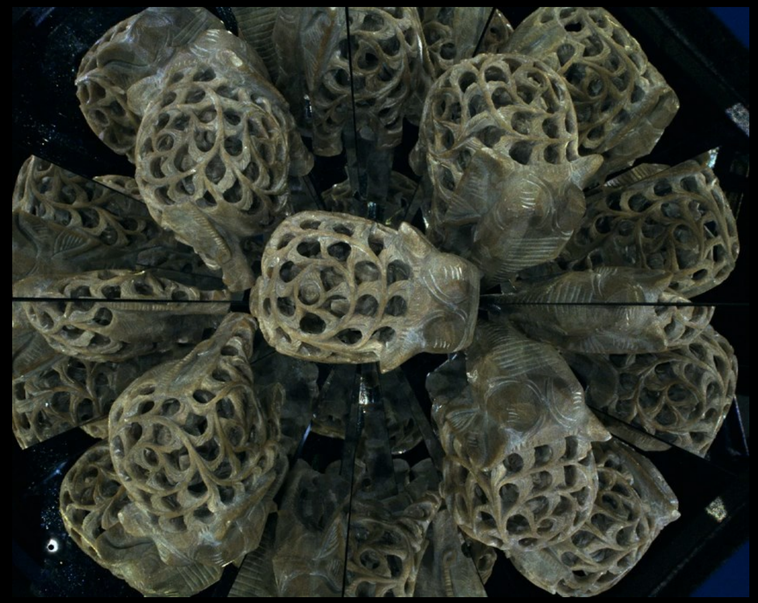


camera view

[Ahn et al., CVPR 2023, Tuesday PM]

Kaleidoscopic 3D scanning

kaleidoscopic system

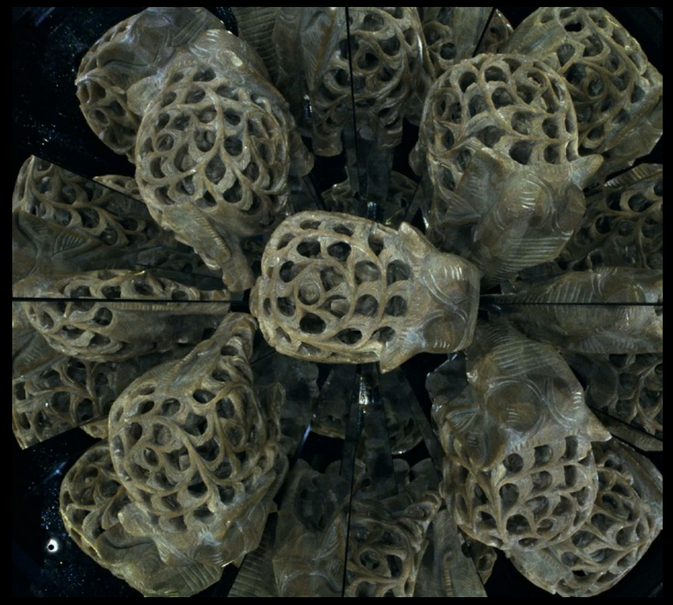


camera view

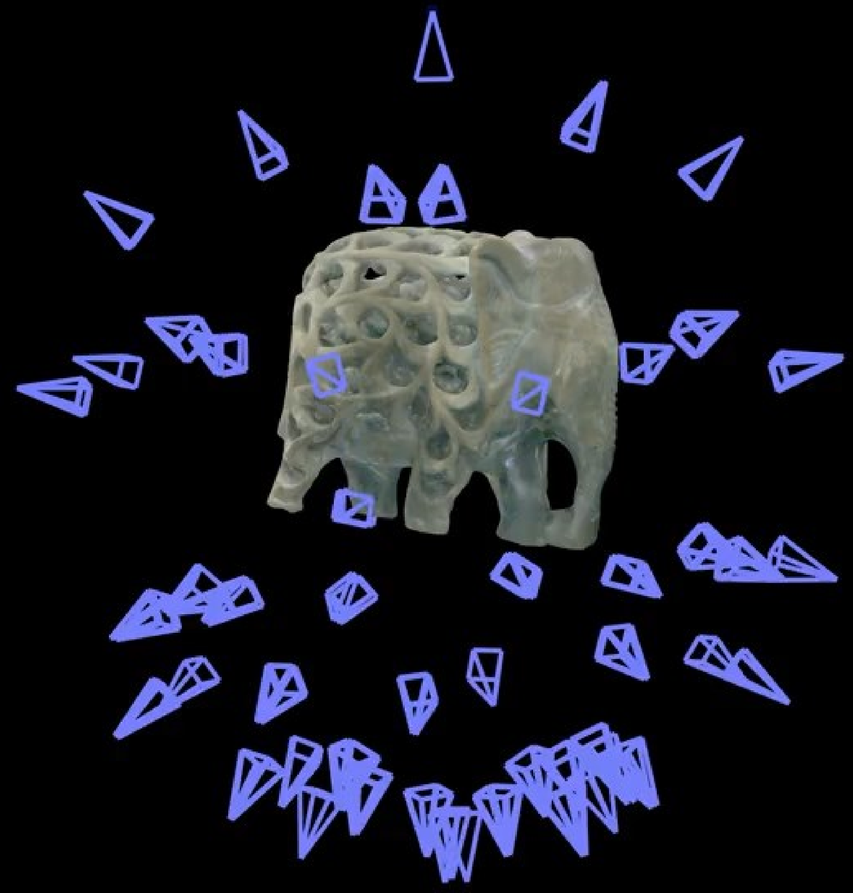
[Ahn et al., CVPR 2023, Tuesday PM]

Kaleidoscopic 3D scanning

kaleidoscopic system



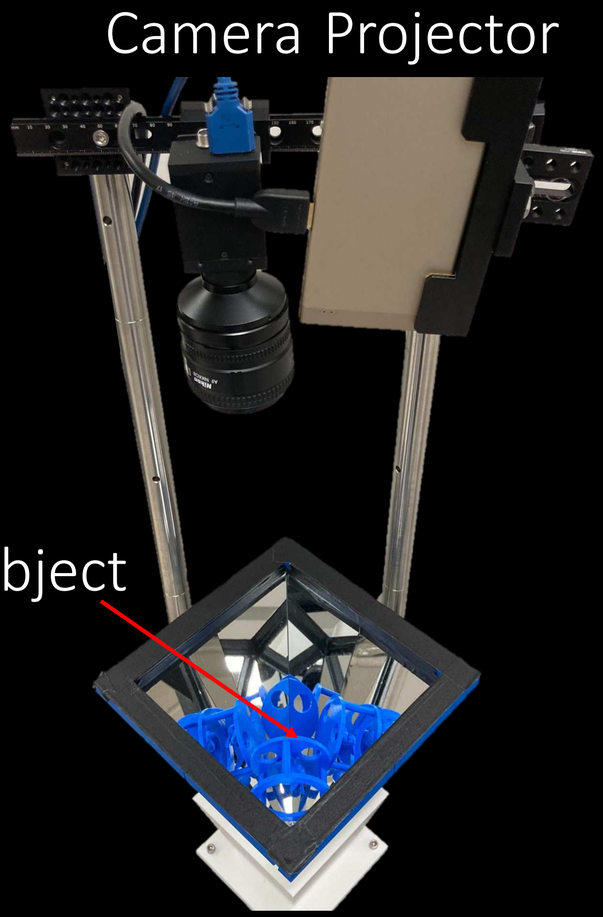
camera view



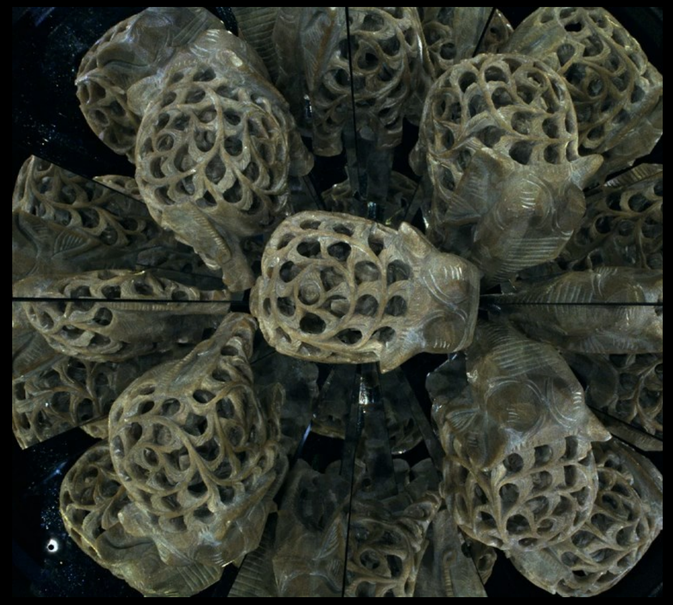
virtual cameras

[Ahn et al., CVPR 2023, Tuesday PM]

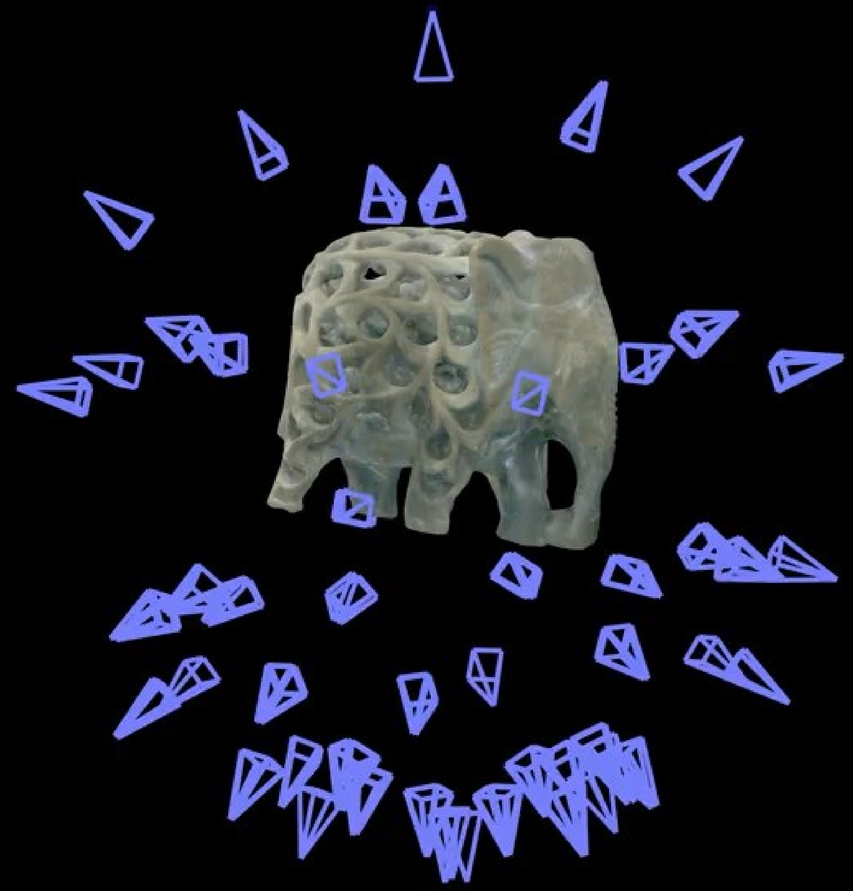
Kaleidoscopic 3D scanning



kaleidoscopic system



camera view

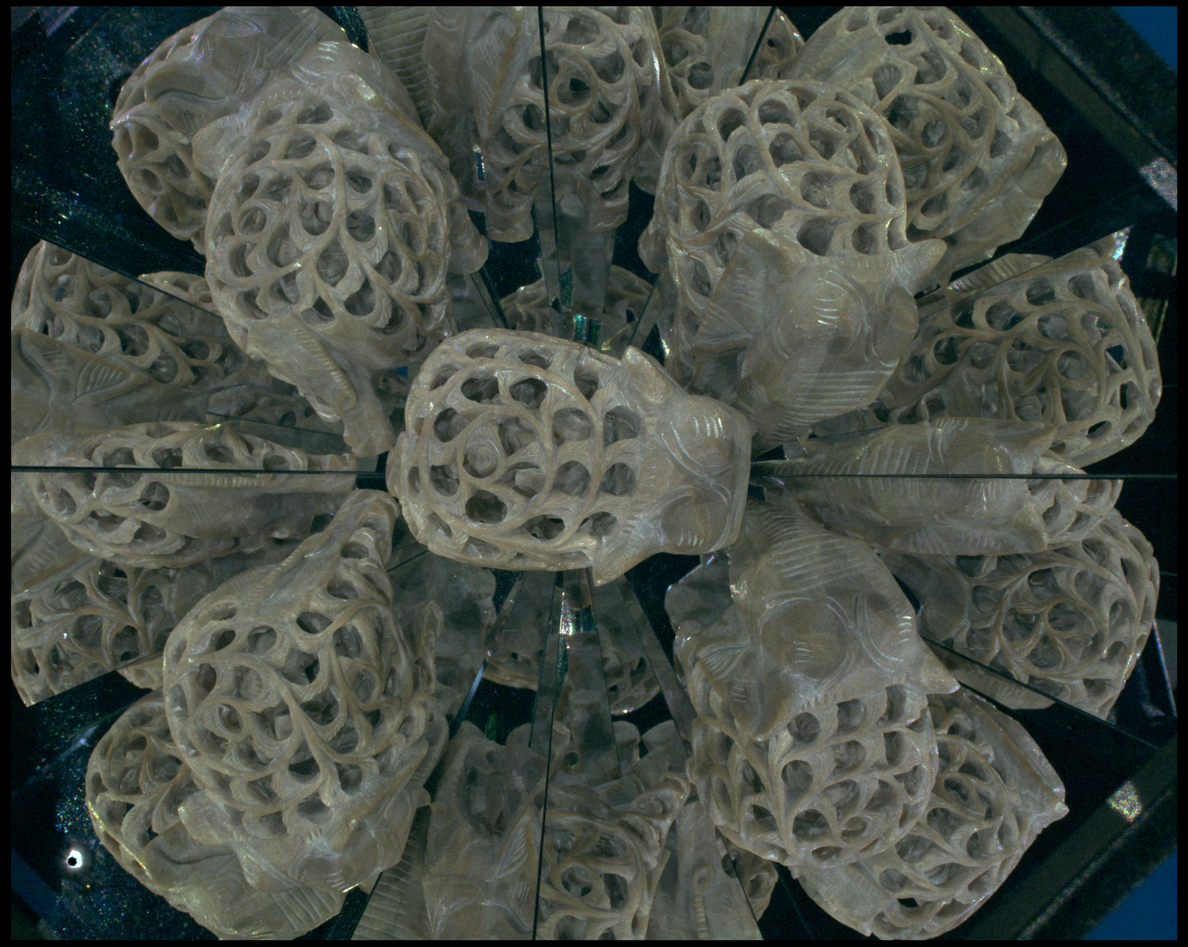


virtual cameras

[Ahn et al., CVPR 2023, Tuesday PM]

Example 3D scans

photograph

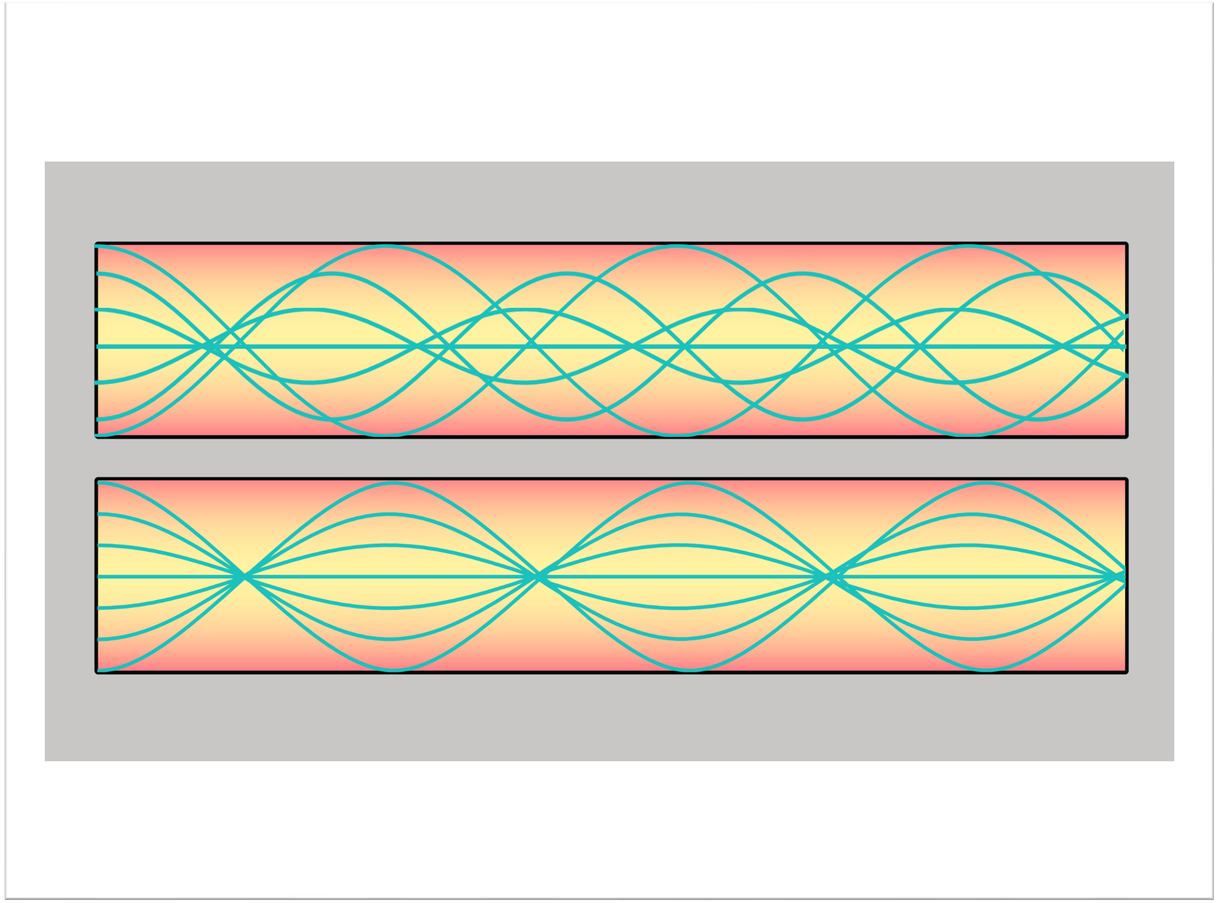


3D reconstruction

[Ahn et al., CVPR 2023, Tuesday PM]

Optimizing Gradient-Index (GRIN) Optics

Luneburg Lens



GRIN Fiber

[Teh et al., 2022]

Nonlinear Ray Tracing

$\eta(\mathbf{x})$: refractive index of the volume at location, $\mathbf{x}$

[Teh et al., 2022]

Nonlinear Ray Tracing

[Teh et al., 2022]

Nonlinear Ray Tracing

[Teh et al., 2022]

Nonlinear Ray Tracing

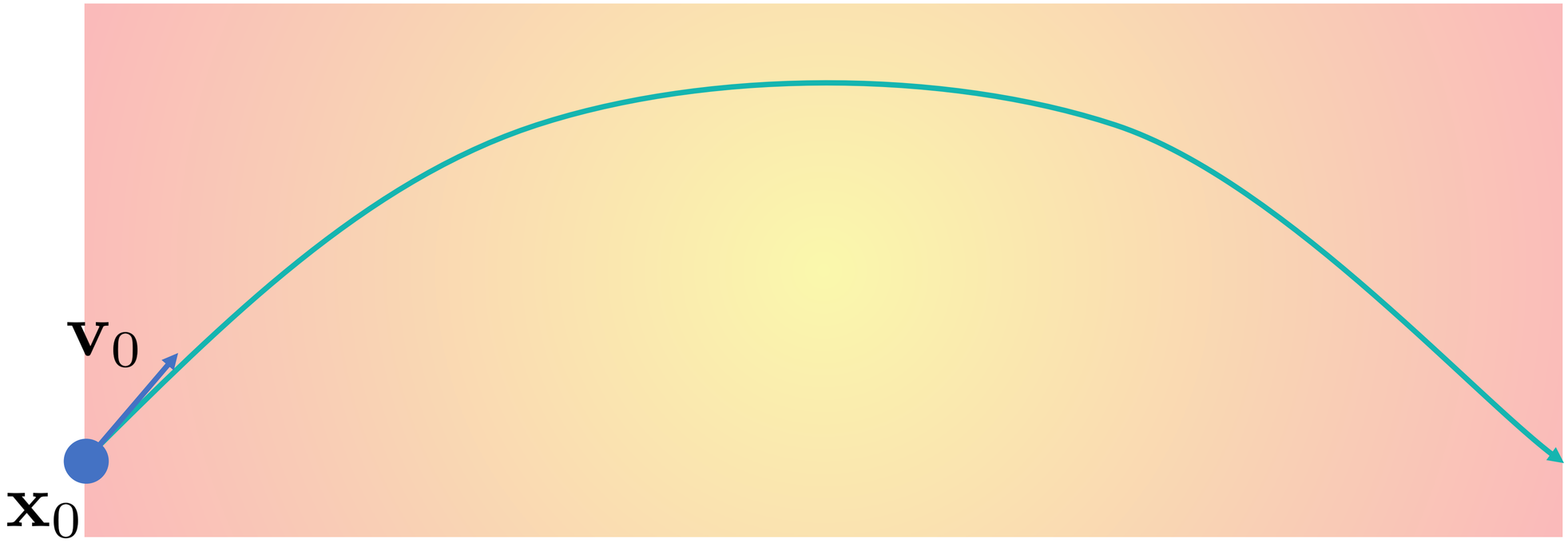
[Teh et al., 2022]

Nonlinear Ray Tracing

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{v} \qquad \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = \eta\nabla\eta$$

[Teh et al., 2022]

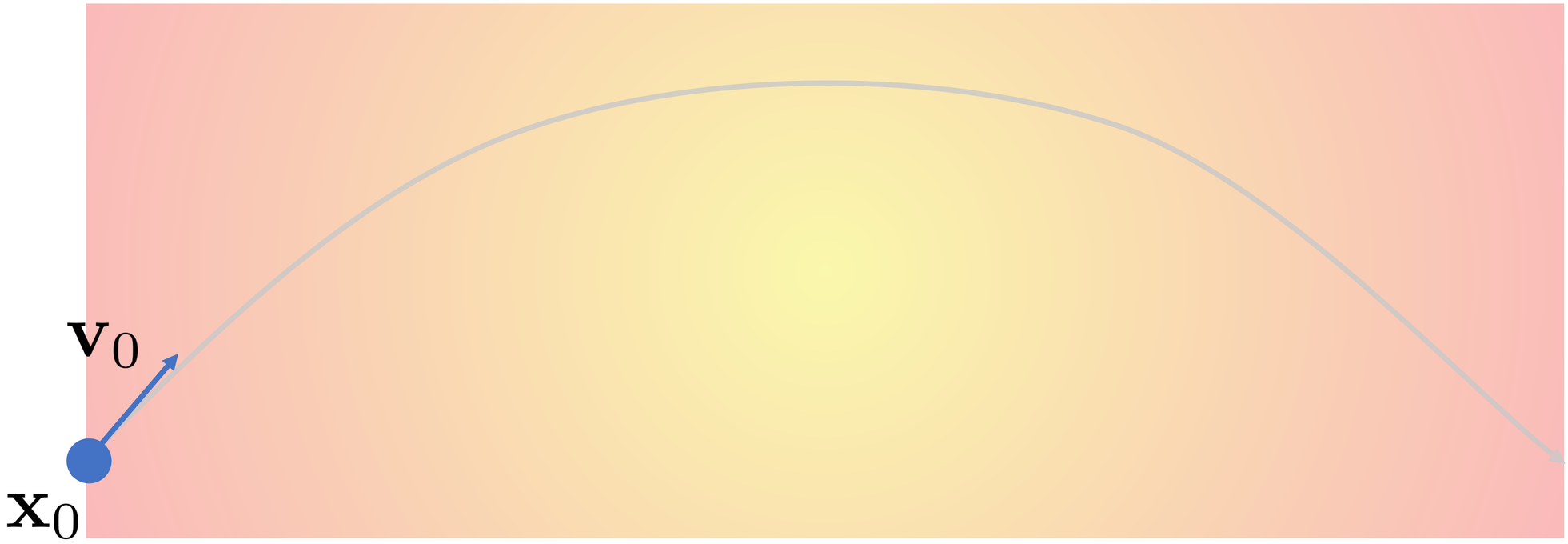
Nonlinear Ray Tracing



$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{v} \qquad \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = \eta\nabla\eta$$

[Teh et al., 2022]

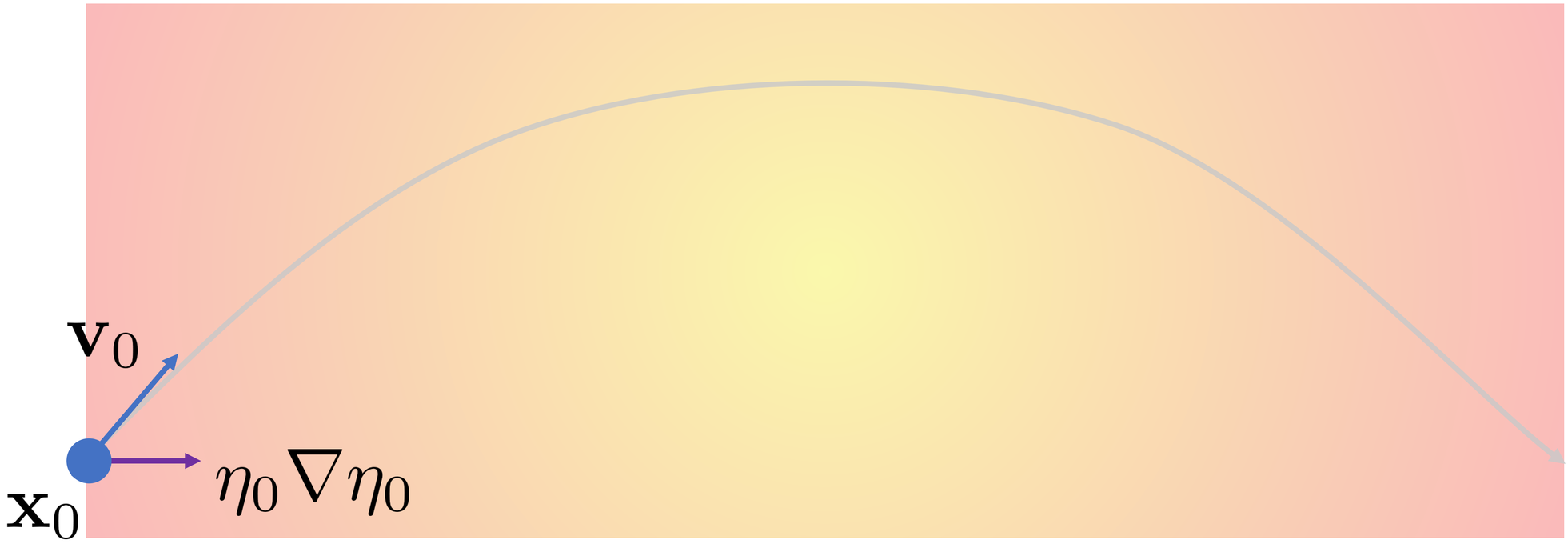
Nonlinear Ray Tracing



$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t \qquad \mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

[Teh et al., 2022]

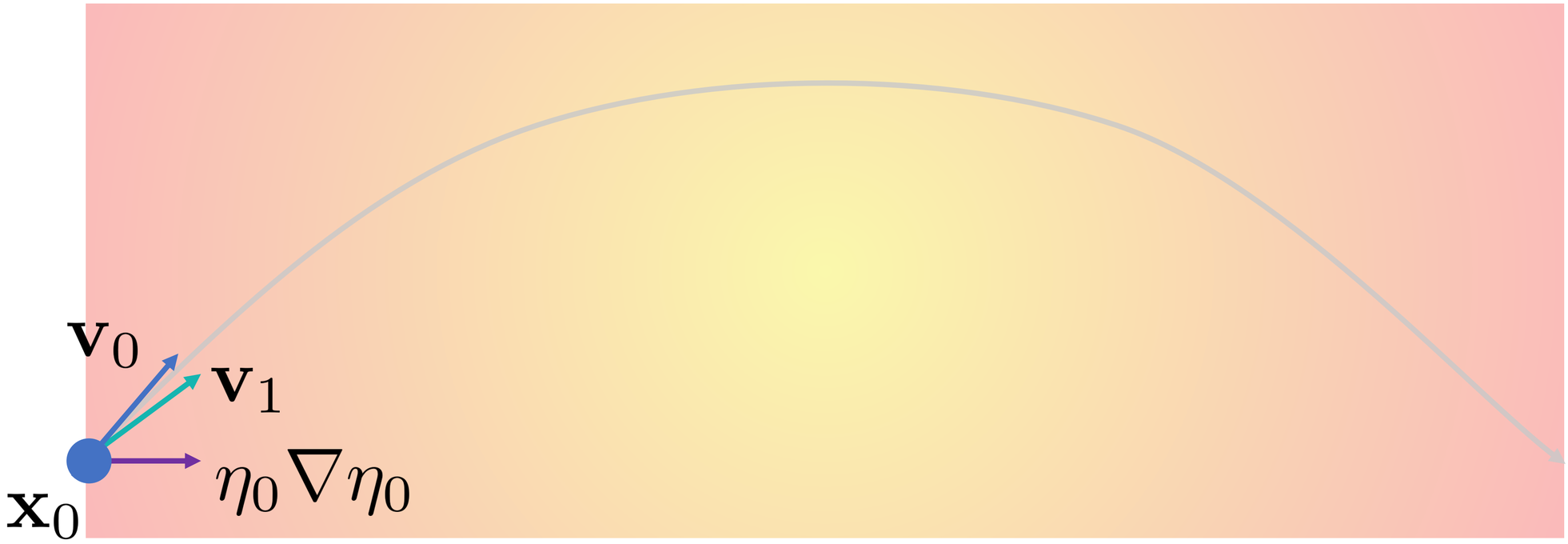
Nonlinear Ray Tracing



$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t \qquad \mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

[Teh et al., 2022]

Nonlinear Ray Tracing

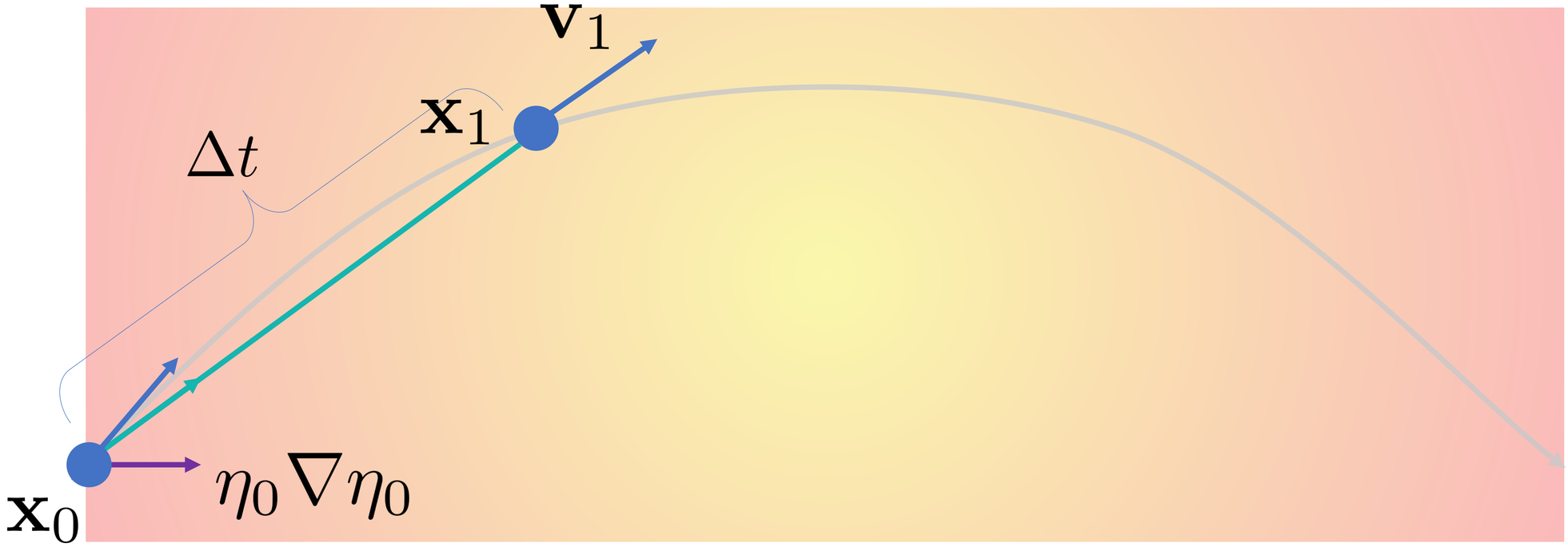


$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t$$

$$\mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

[Teh et al., 2022]

Nonlinear Ray Tracing

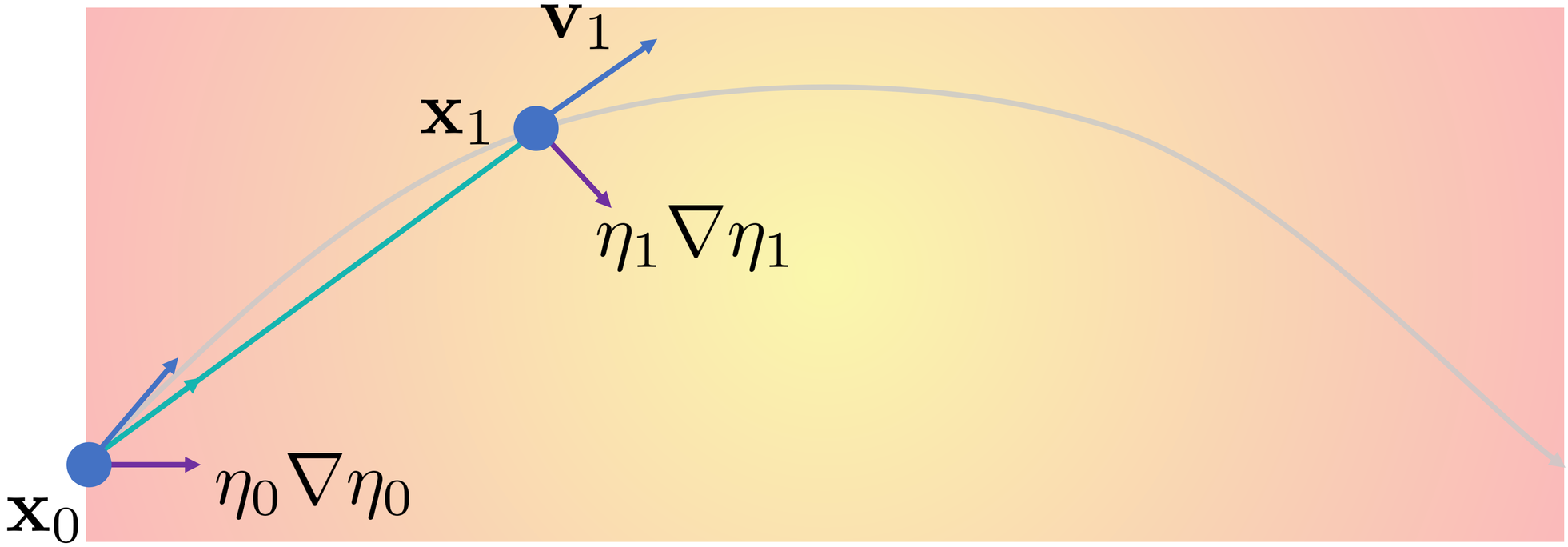


$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t$$

$$\mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

[Teh et al., 2022]

Nonlinear Ray Tracing

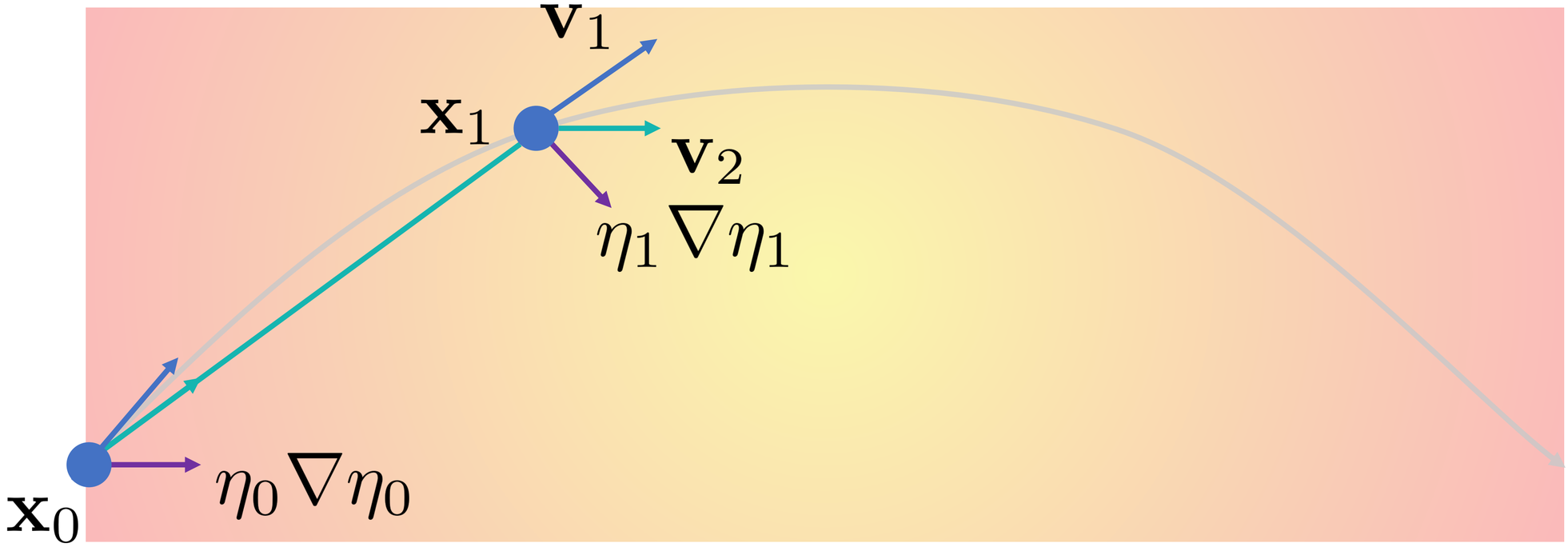


$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t$$

$$\mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

[Teh et al., 2022]

Nonlinear Ray Tracing

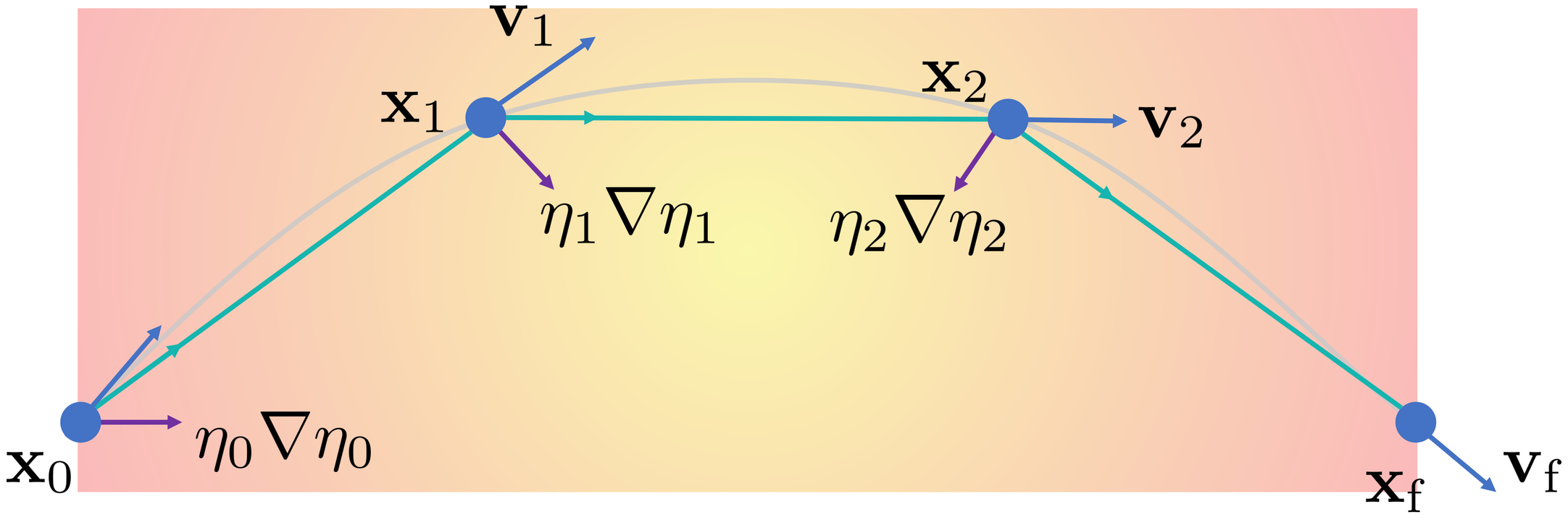


$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t$$

$$\mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

[Teh et al., 2022]

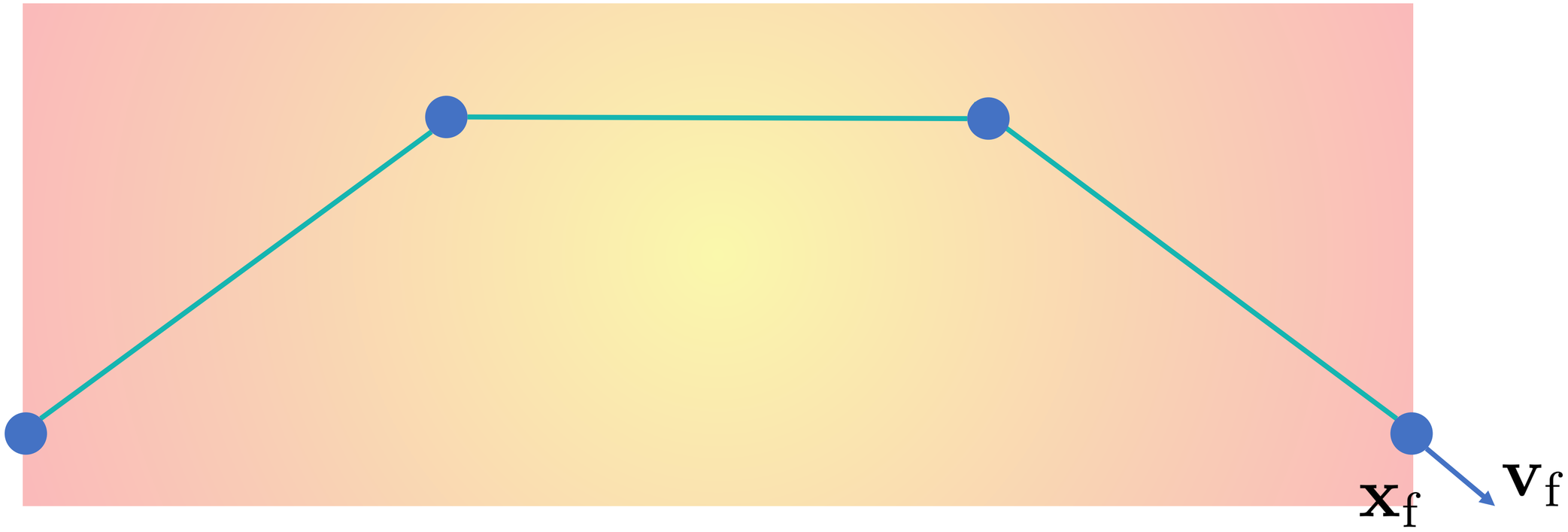
Nonlinear Ray Tracing



$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{v}_i \Delta t \qquad \mathbf{v}_i = \mathbf{v}_{i-1} + \eta_{i-1} \nabla \eta_{i-1} \Delta t$$

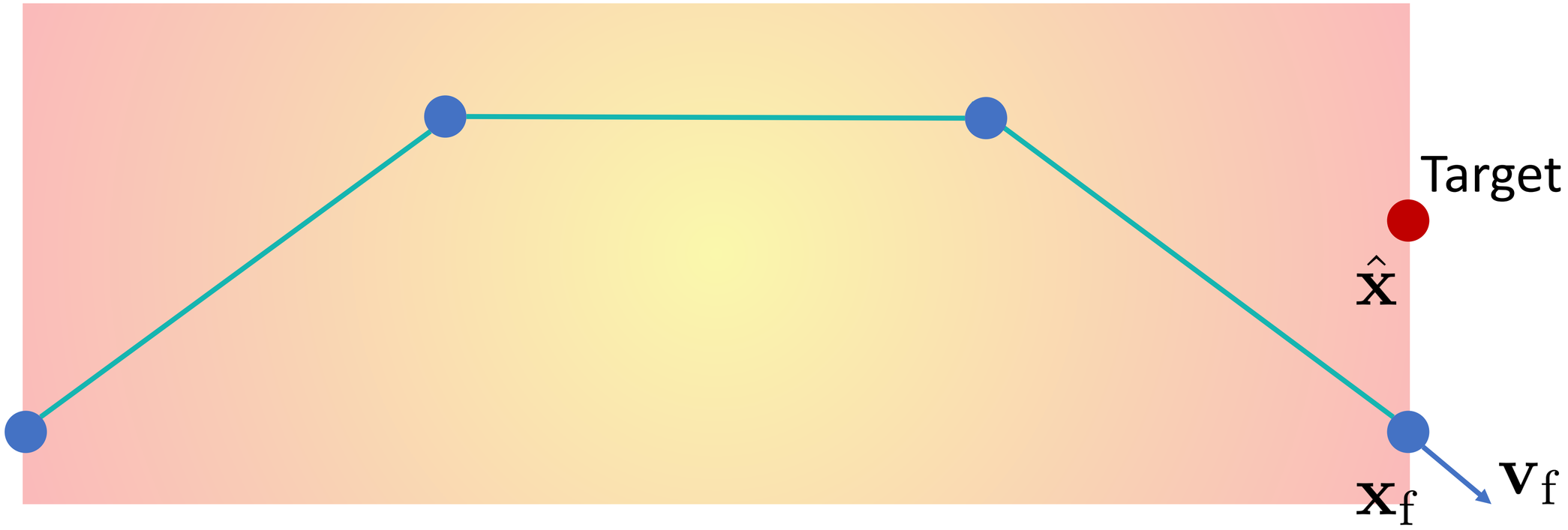
[Teh et al., 2022]

Nonlinear Ray Tracing



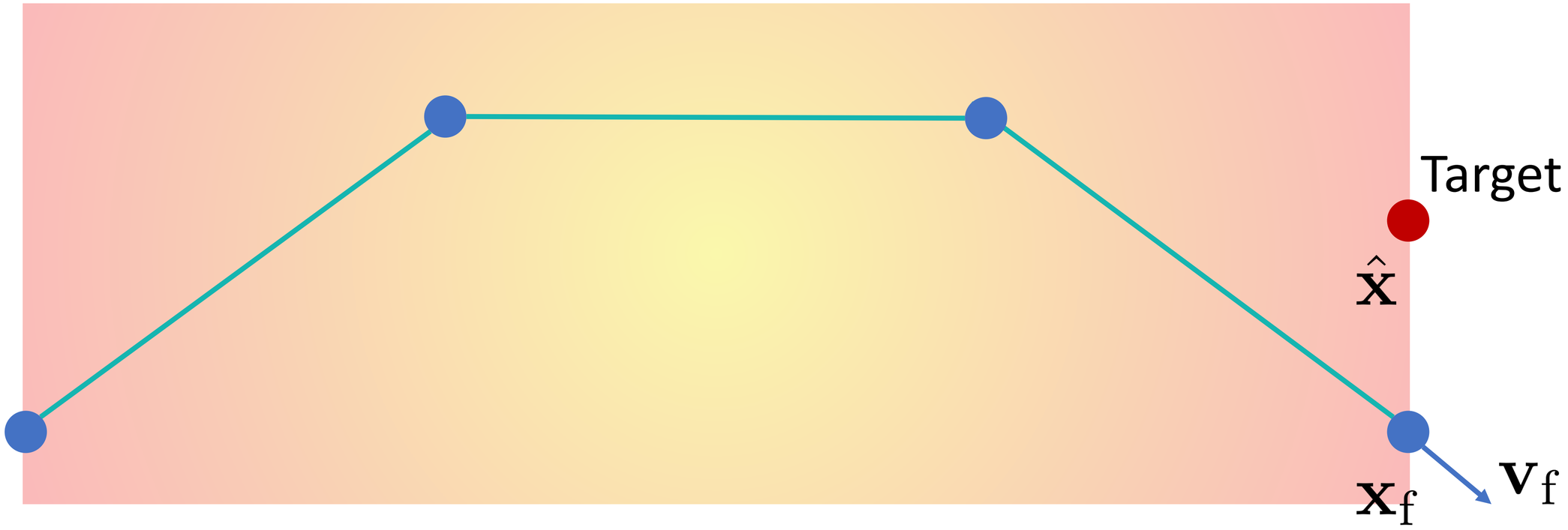
[Teh et al., 2022]

Nonlinear Ray Tracing



[Teh et al., 2022]

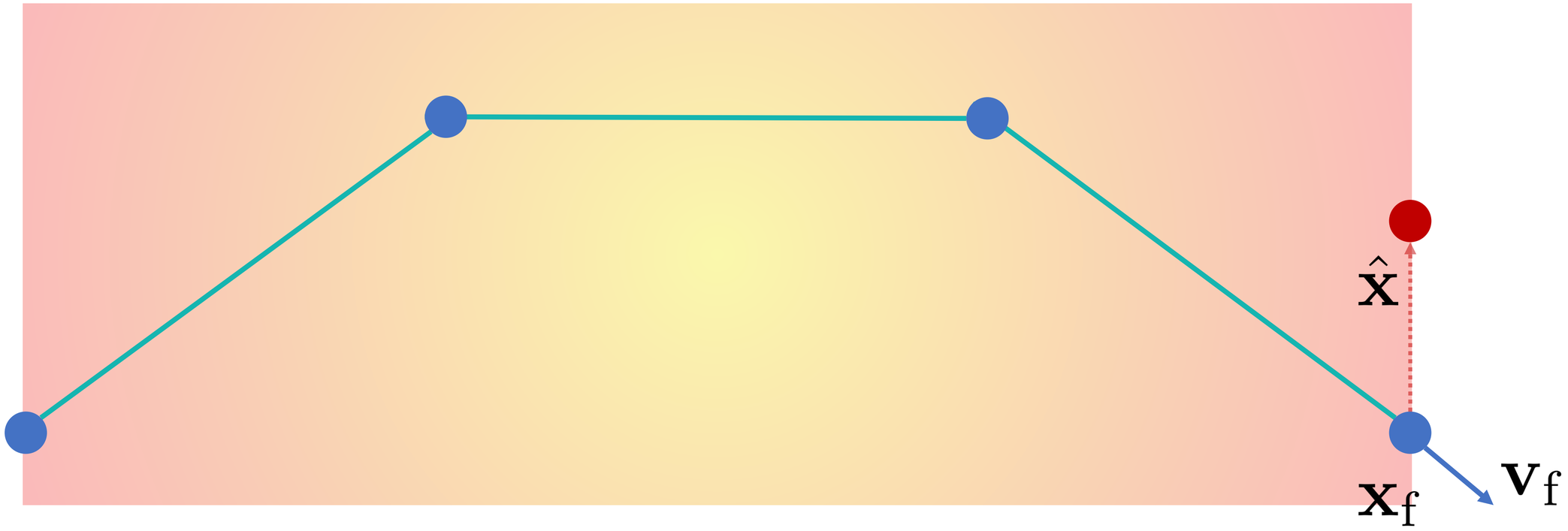
Nonlinear Ray Tracing



$$\min_{\eta} \left\| \hat{\mathbf{x}} - \mathbf{x}_\mathrm{f} \right\|^2$$

[Teh et al., 2022]

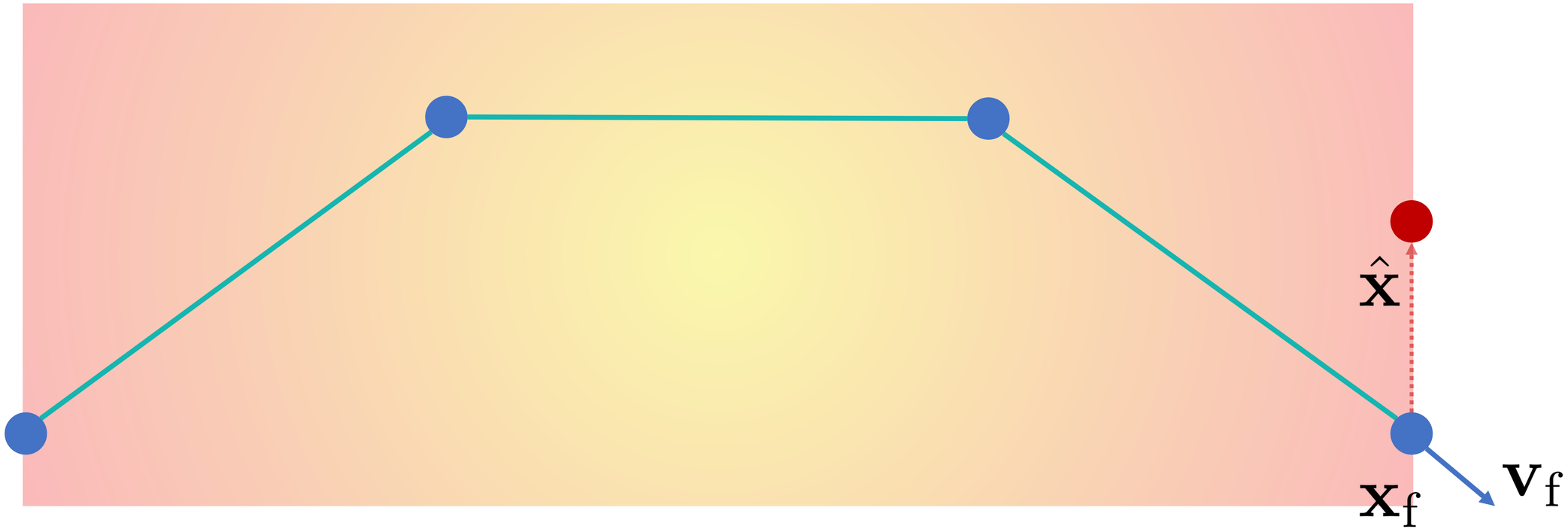
Nonlinear Ray Tracing



$$\frac{\mathrm{d}}{\mathrm{d}\eta}\left\|\hat{\mathbf{x}}-\mathbf{x}_\mathrm{f}\right\|^2$$

[Teh et al., 2022]

Nonlinear Ray Tracing in reverse



$\mathbf{x}_0$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_\mathrm{f}$
$\mathbf{v}_0$	$\mathbf{v}_1$	$\mathbf{v}_2$	$\mathbf{v}_\mathrm{f}$
$\eta\nabla\eta$	$\eta\nabla\eta$	$\eta\nabla\eta$	$\eta\nabla\eta$

[Teh et al., 2022]

Nonlinear Ray Tracing in reverse

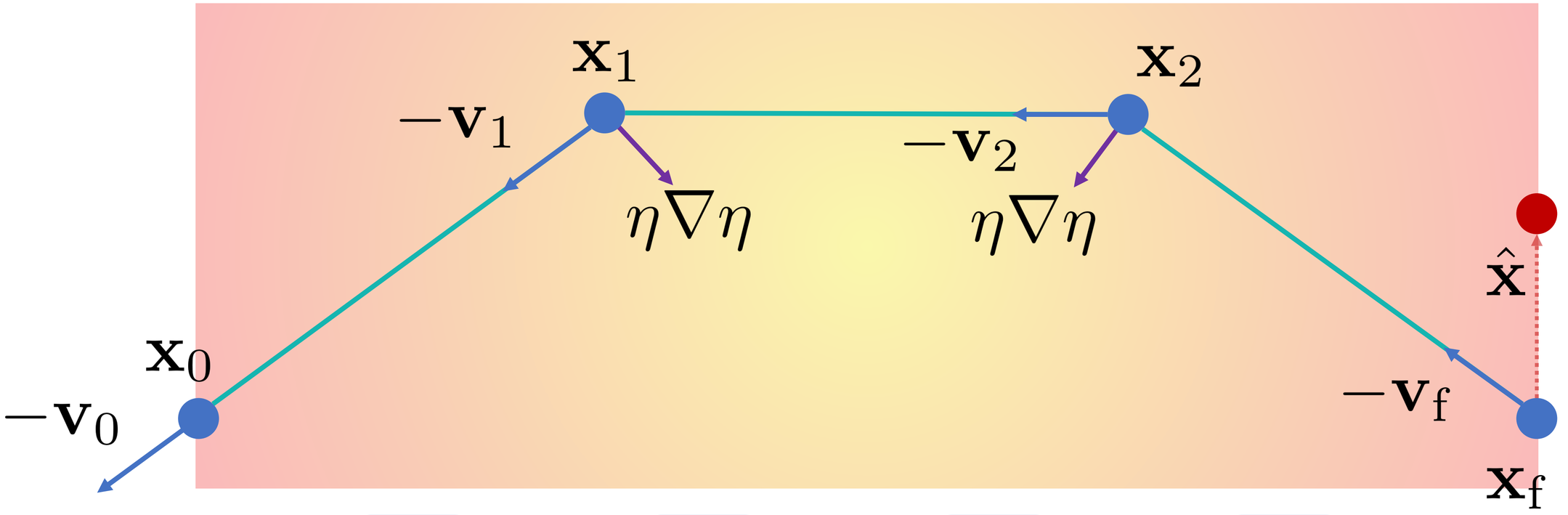
$\mathbf{x}_\mathrm{f}$

$\mathbf{v}_\mathrm{f}$

$\eta\nabla\eta$

[Teh et al., 2022]

Nonlinear Ray Tracing in reverse



$\mathbf{x}_0$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_\mathrm{f}$
$\mathbf{v}_0$	$\mathbf{v}_1$	$\mathbf{v}_2$	$\mathbf{v}_\mathrm{f}$
$\eta\nabla\eta$	$\eta\nabla\eta$	$\eta\nabla\eta$	$\eta\nabla\eta$

[Teh et al., 2022]

Nonlinear Ray Tracing in reverse

$$\min_{\eta} \sum_{i=1}^{N} \mathcal{F}_i \left[\iint_{(\mathbf{x}_0, \mathbf{v}_0) \in \Omega} C_i \left(\mathbf{x} \left(\sigma_f; \eta, \mathbf{x}_0, \mathbf{v}_0 \right), \mathbf{v} \left(\sigma_f; \eta, \mathbf{x}_0, \mathbf{v}_0 \right) \right) \, \mathrm{d}\mathbf{x}_0 \, \mathrm{d}\mathbf{v}_0 \right]$$

$$\text{s.t. } \dot{\mathbf{x}} \left(\sigma; \eta, \mathbf{x}_0, \mathbf{v}_0 \right) = \mathbf{v}, \quad \forall \sigma \in \left[0, \sigma_f \right],$$

$$\dot{\mathbf{v}} \left(\sigma; \eta, \mathbf{x}_0, \mathbf{v}_0 \right) = \eta \nabla \eta, \quad \forall \sigma \in \left[0, \sigma_f \right],$$

$$\mathbf{x} \left(0; \eta, \mathbf{x}_0, \mathbf{v}_0 \right) = \mathbf{x}_0,$$

$$\mathbf{v} \left(0; \eta, \mathbf{x}_0, \mathbf{v}_0 \right) = \mathbf{v}_0, \tag{15}$$

$\mathbf{x}_0$

$\hat{\mathbf{x}}$

$$\dot{\boldsymbol{\lambda}} = - \left(\nabla \eta \left(\nabla \eta \right)^{\top} + \eta \, \mathrm{Hess} \left(\eta \right) \right) \boldsymbol{\mu}, \qquad \forall \sigma \in \left[0, \sigma_f \right] \tag{19}$$

$$\dot{\boldsymbol{\mu}} = -\boldsymbol{\lambda}, \qquad \forall \sigma \in \left[0, \sigma_f \right] \tag{20}$$

$$\boldsymbol{\lambda} \left(\sigma_f \right) = \frac{\partial C}{\partial \mathbf{x}}, \tag{21}$$

$$\boldsymbol{\mu} \left(\sigma_f \right) = \frac{\partial C}{\partial \mathbf{v}}. \tag{22}$$

$$\mathbf{x}_{i-1} = \mathbf{x}_i - \mathbf{v}_i \Delta\sigma, \tag{28}$$

$$\mathbf{v}_{i-1} = \mathbf{v}_i - \eta \left(\mathbf{x}_{i-1} \right) \nabla \eta \left(\mathbf{x}_{i-1} \right) \Delta\sigma, \tag{29}$$

$$\boldsymbol{\lambda}_{i-1} = \boldsymbol{\lambda}_i + \left(\nabla \eta \left(\mathbf{x}_{i-1} \right) \left(\nabla \eta \left(\mathbf{x}_{i-1} \right) \right)^{\top} + \eta \left(\mathbf{x}_{i-1} \right) \mathrm{Hess} \left(\eta \left(\mathbf{x}_{i-1} \right) \right) \right) \boldsymbol{\mu}_i \Delta\sigma, \tag{30}$$

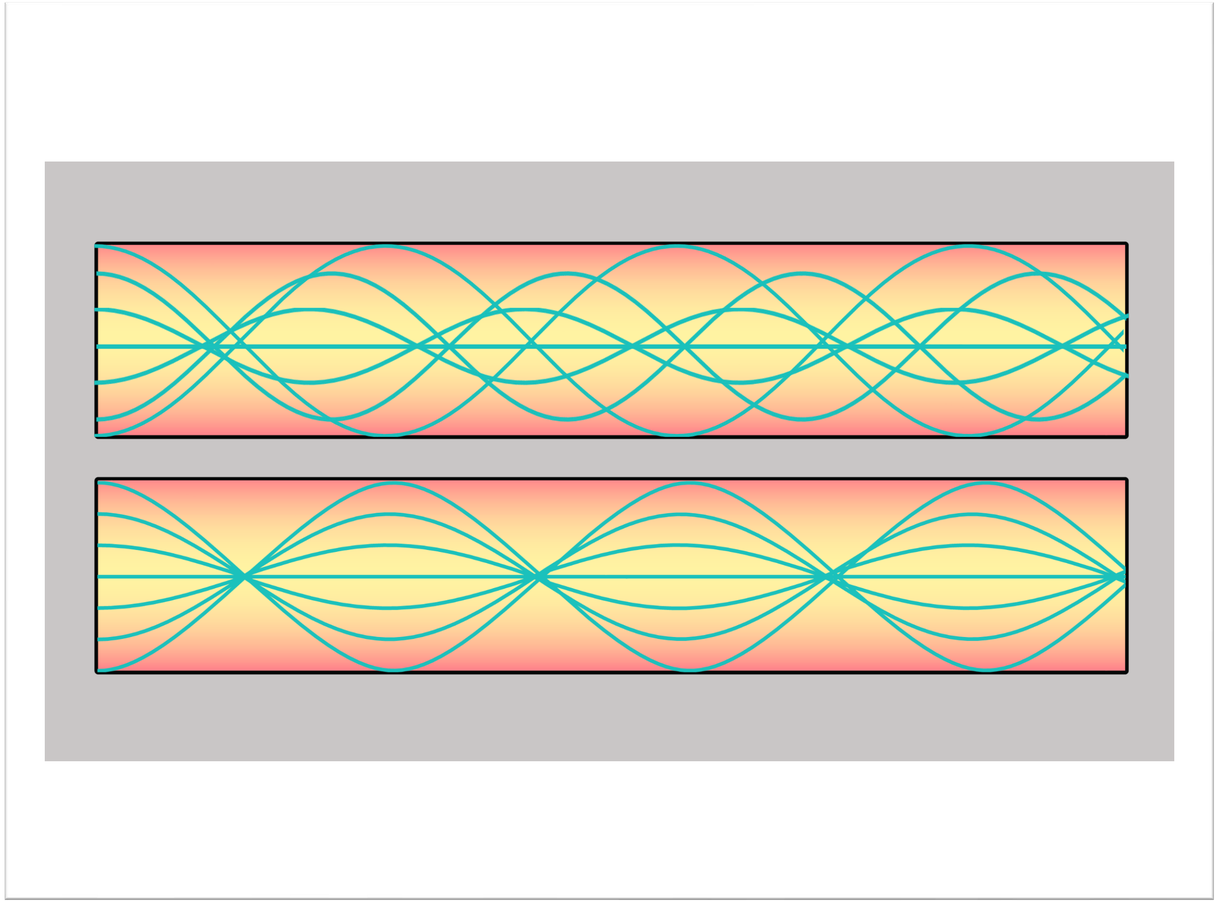
$$\boldsymbol{\mu}_{i-1} = \boldsymbol{\mu}_i + \boldsymbol{\lambda}_{i-1} \Delta\sigma. \tag{31}$$

$\eta\nabla\eta$ $\eta\nabla\eta$ $\eta\nabla\eta$ $\eta\nabla\eta$

[Teh et al., 2022]

Optimizing Gradient-Index (GRIN) Optics

Luneburg Lens

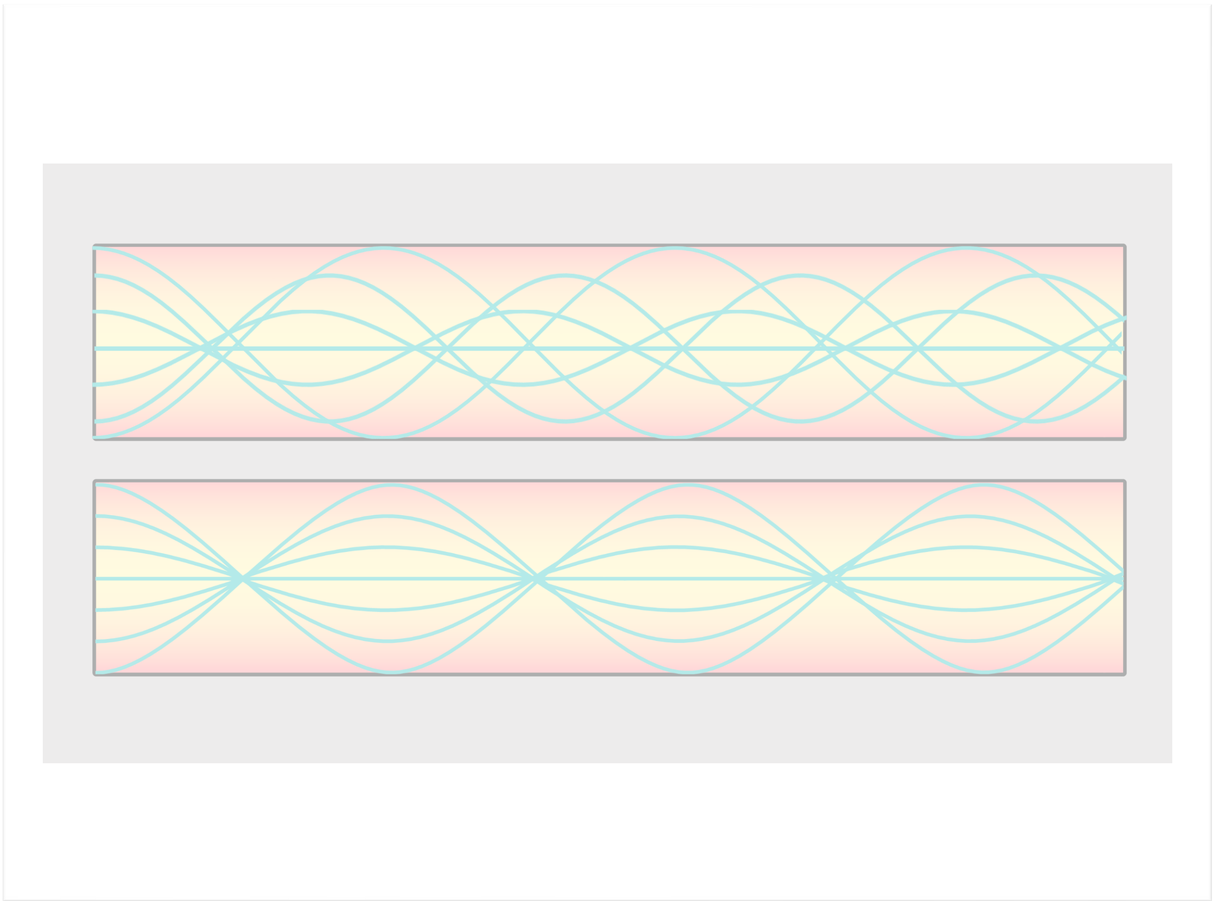


GRIN Fiber

[Teh et al., 2022]

Optimizing Gradient-Index (GRIN) Optics

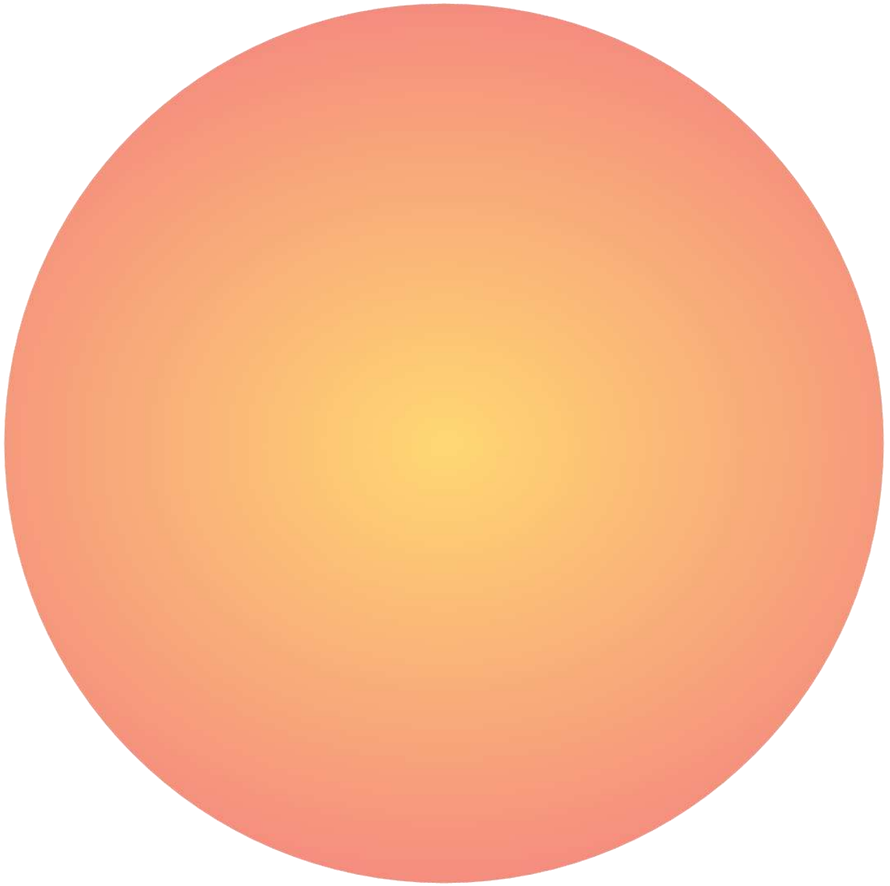
Luneburg Lens



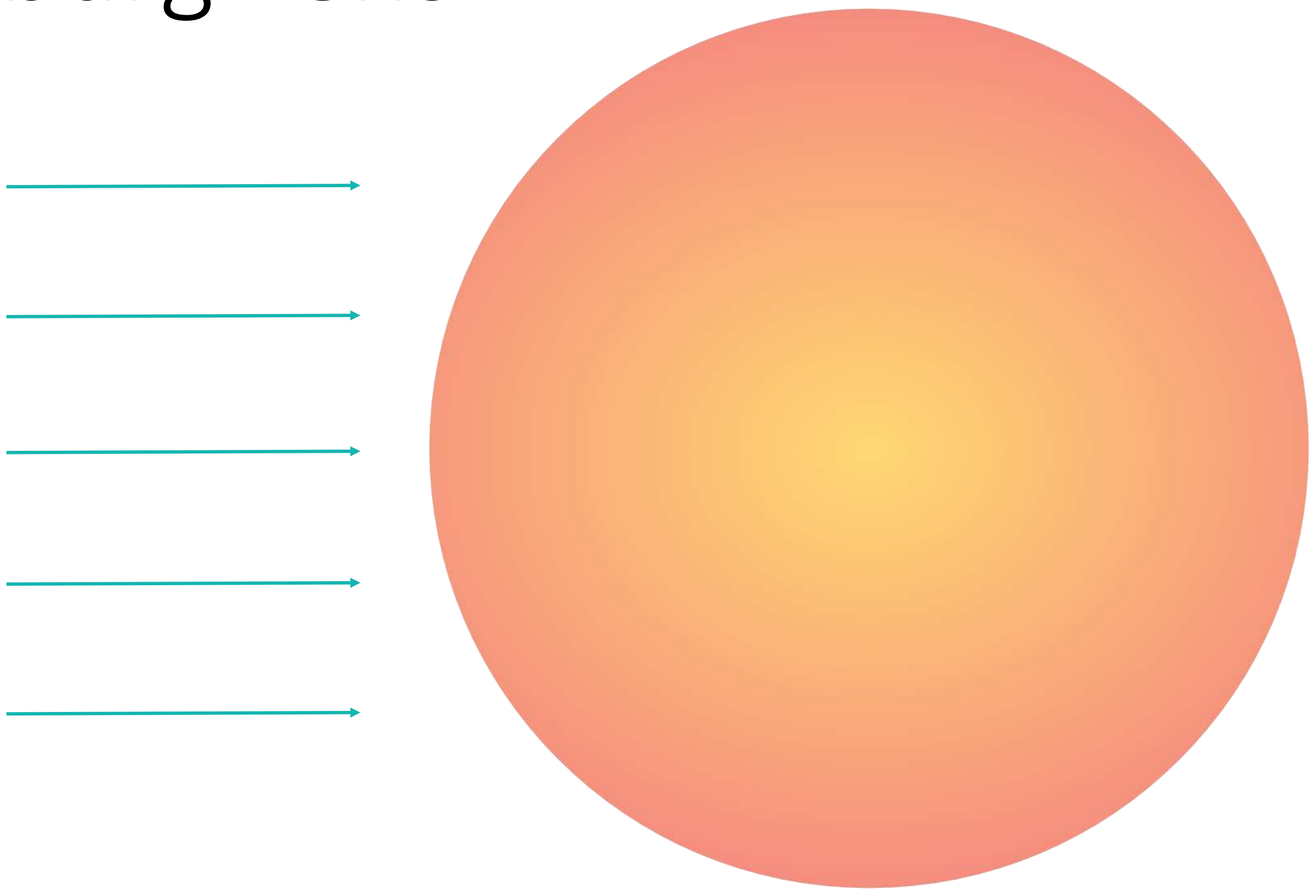
GRIN Fiber

[Teh et al., 2022]

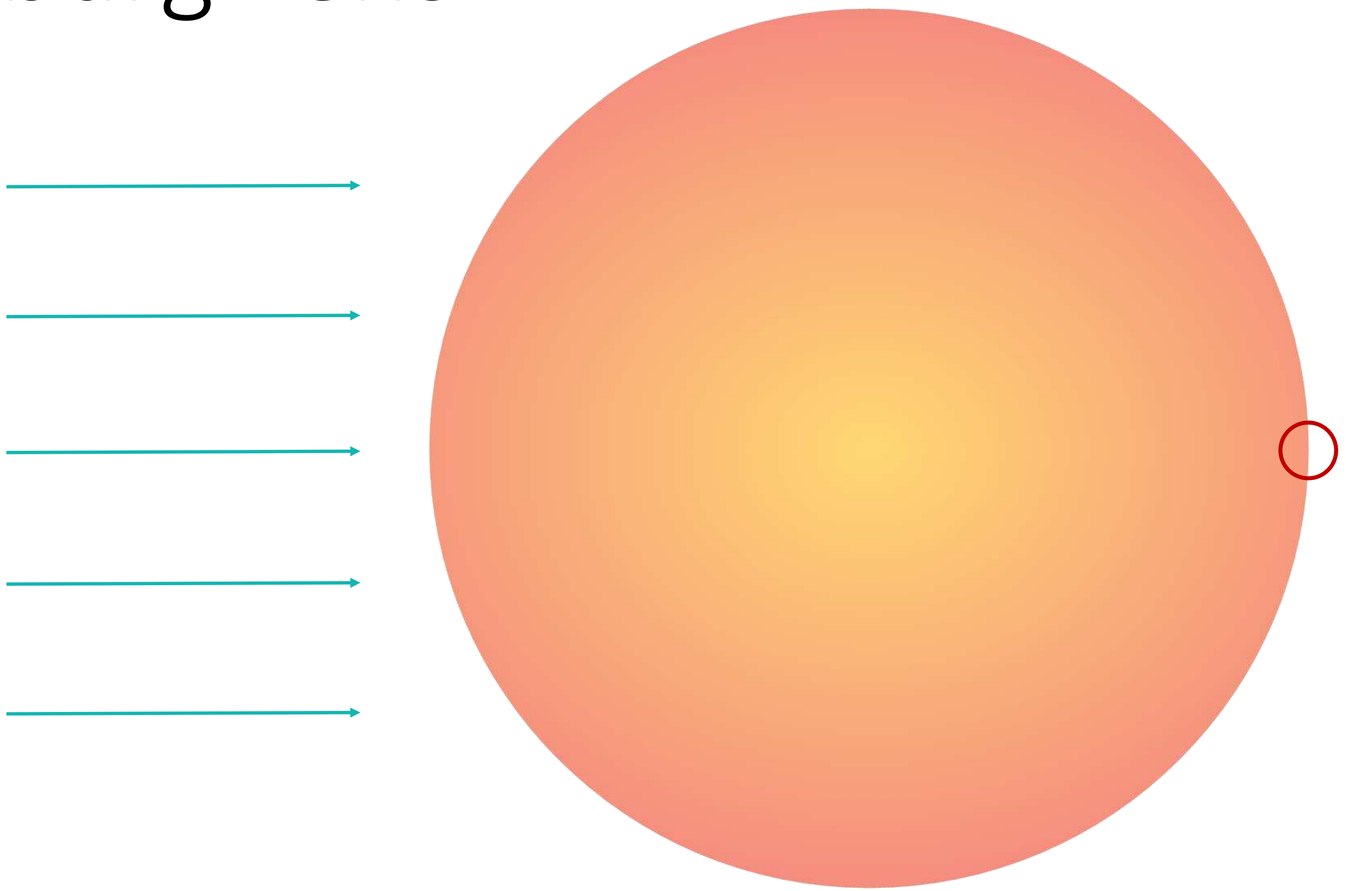
Luneburg Lens



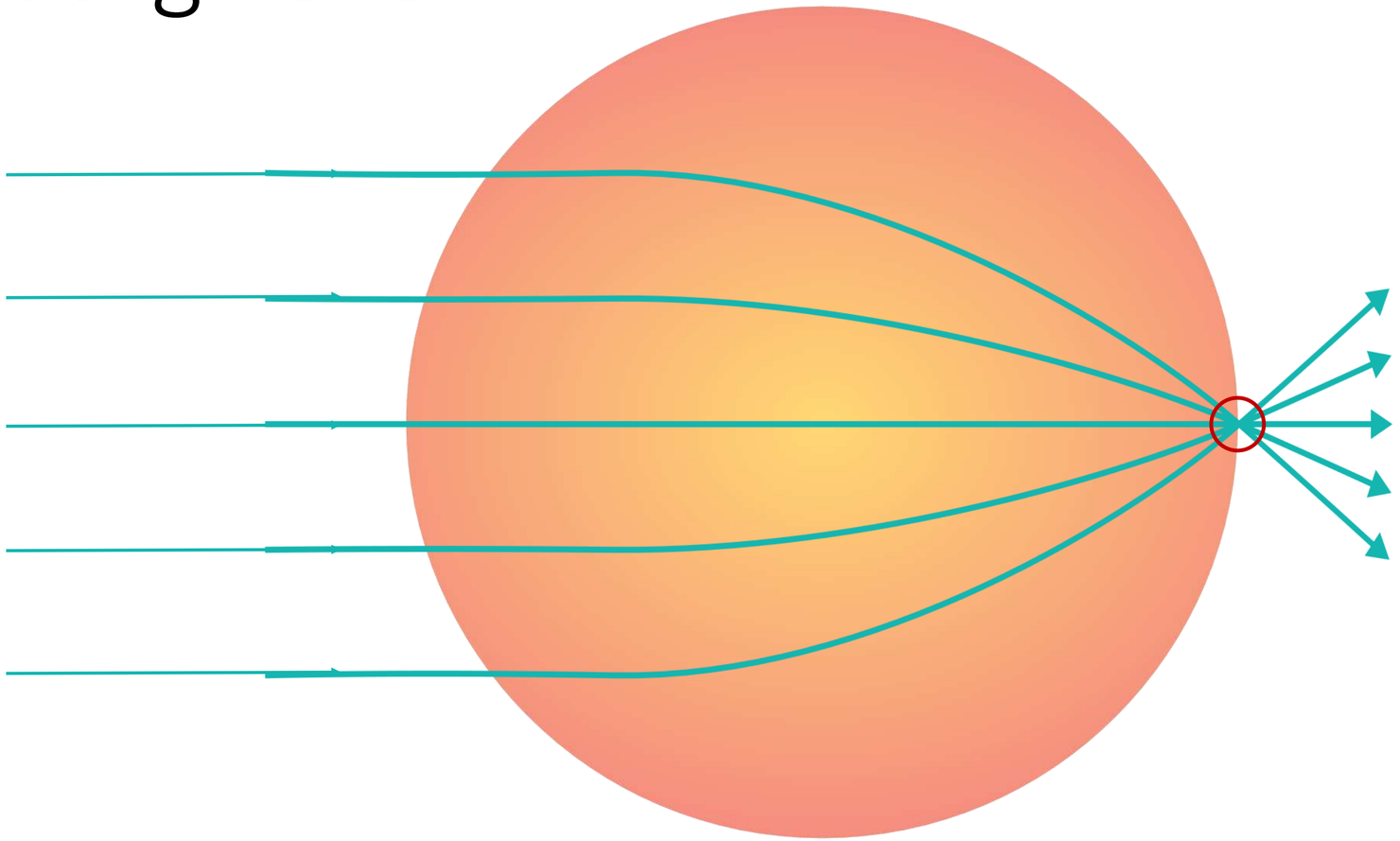
Luneburg Lens



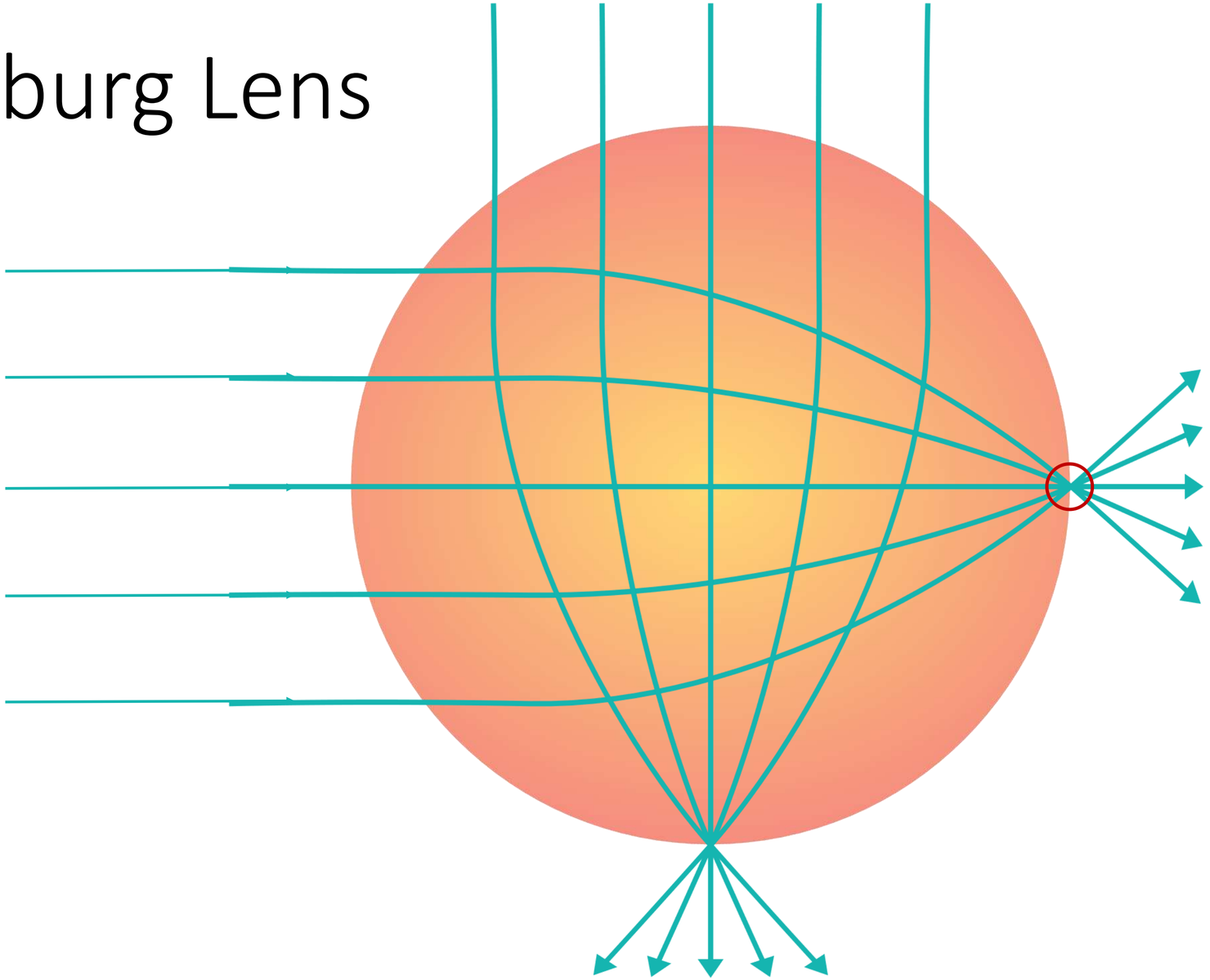
Luneburg Lens



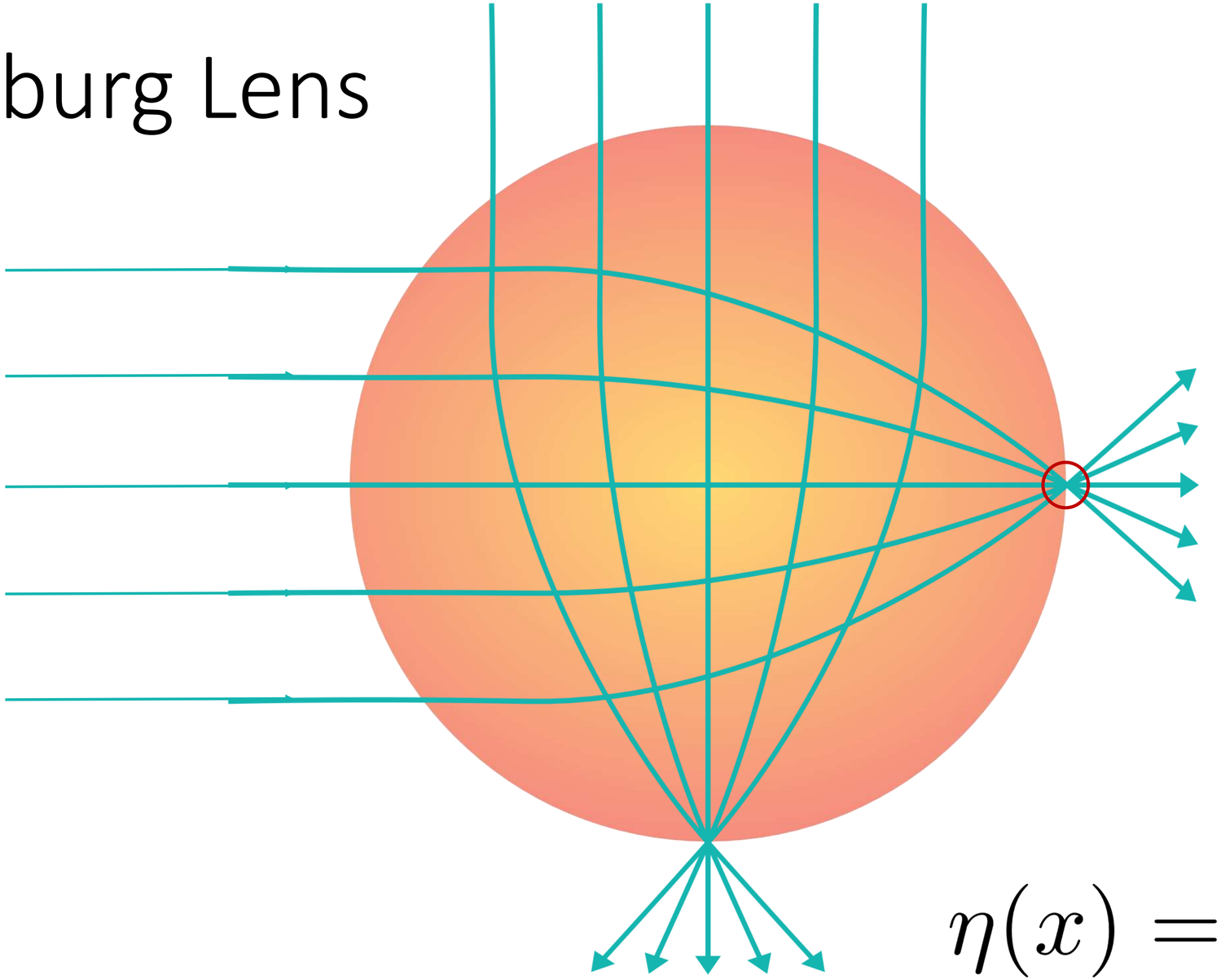
Luneburg Lens



Luneburg Lens



Luneburg Lens



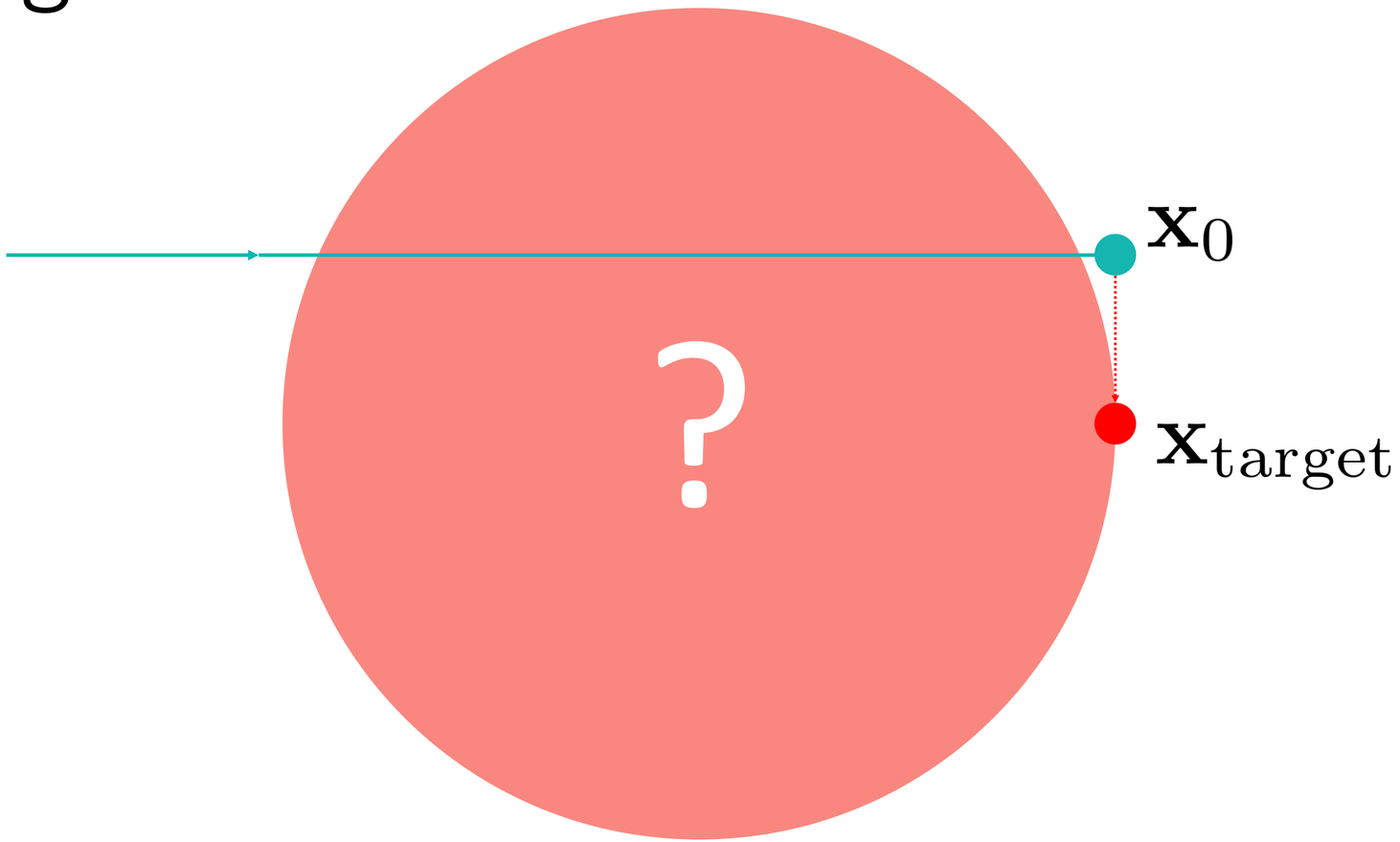
$$\eta(x) = \sqrt{2 - \|x\|^2}$$

[Luneburg, R. K. 1944]

Luneburg Lens

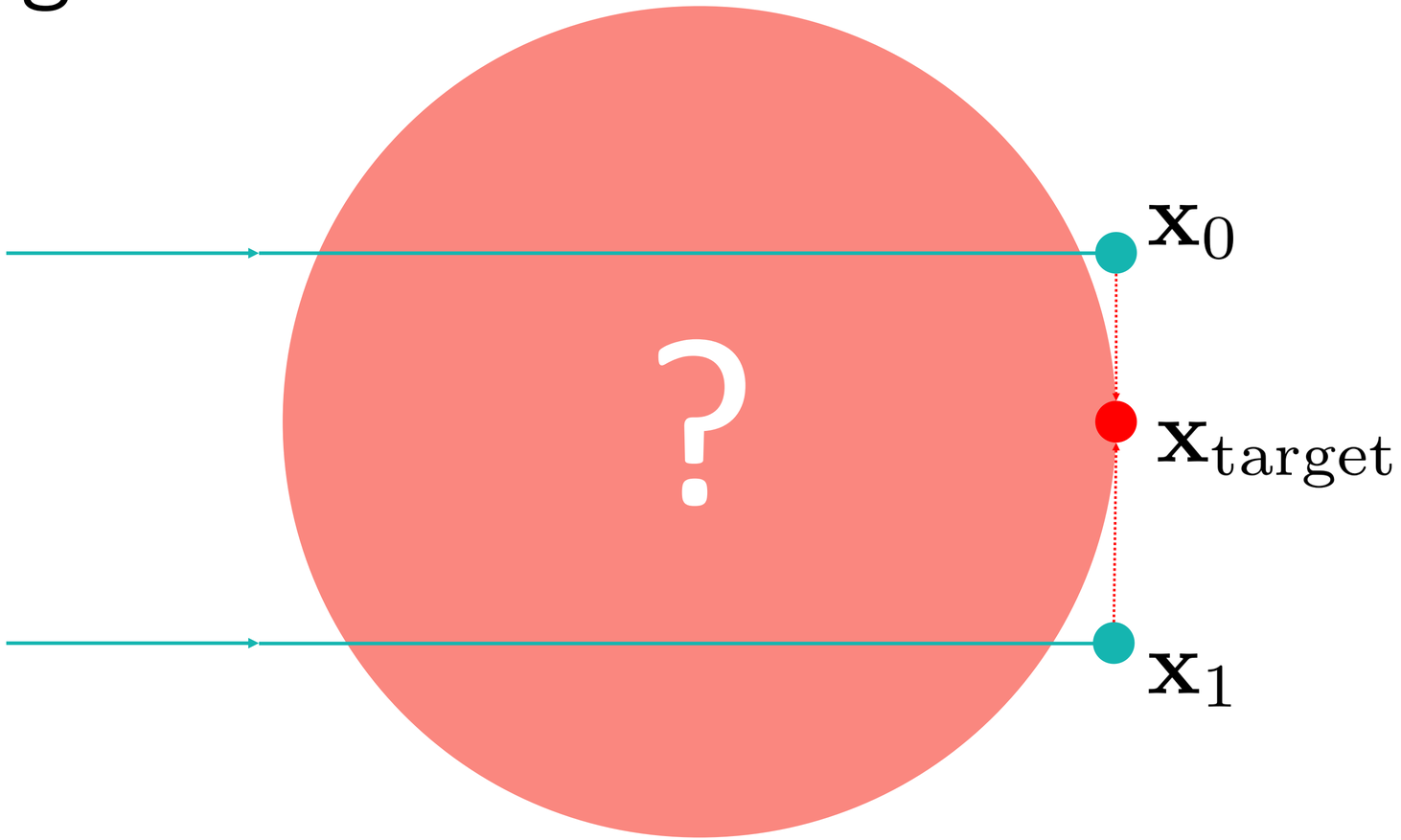
[Teh et al., 2022]

Luneburg Lens



[Teh et al., 2022]

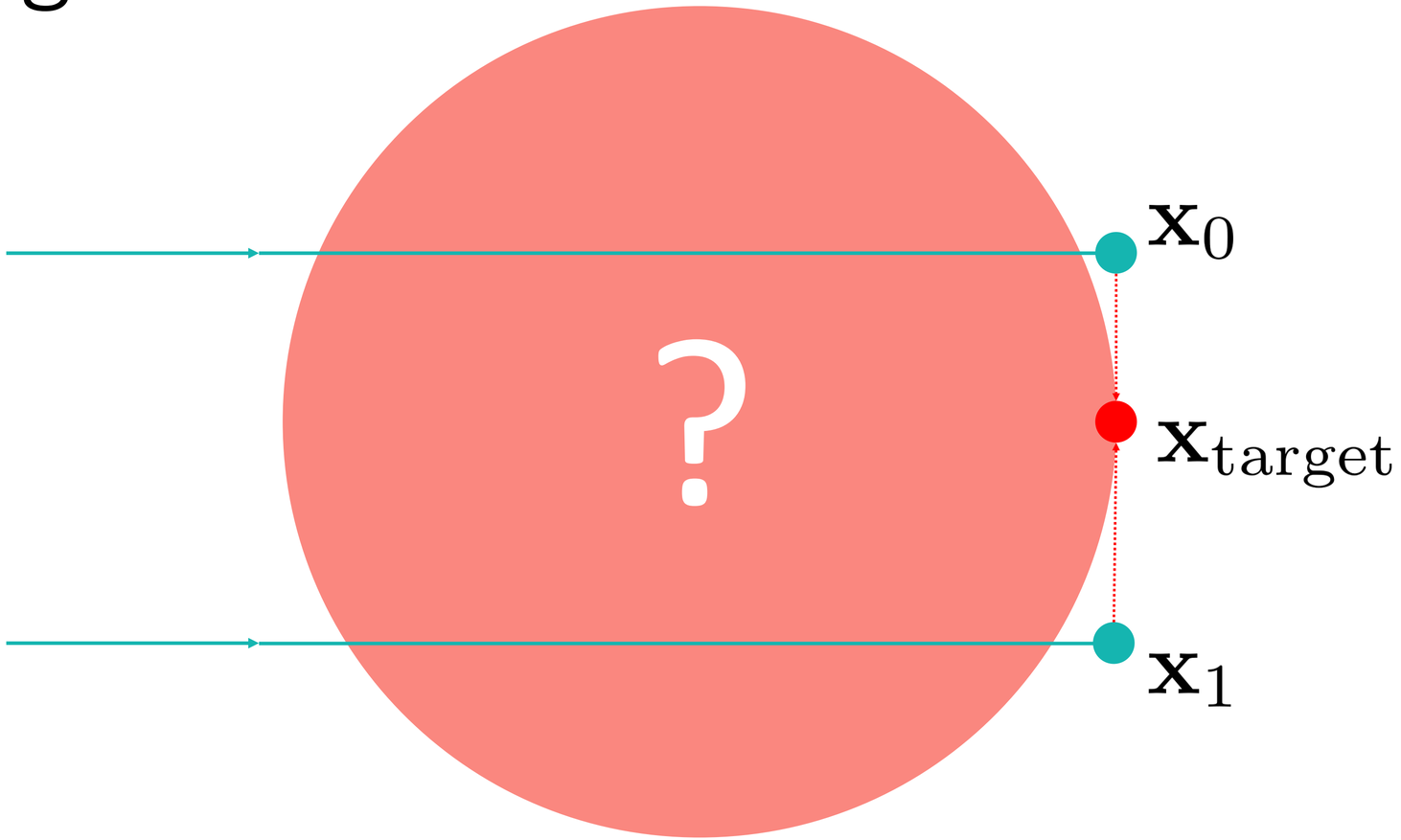
Luneburg Lens



$$\min_{\eta} \sum_{i \in \text{rays}} \left\| \mathbf{x}_{\text{target}} - \mathbf{x}_{\text{i}} \right\|^2$$

[Teh et al., 2022]

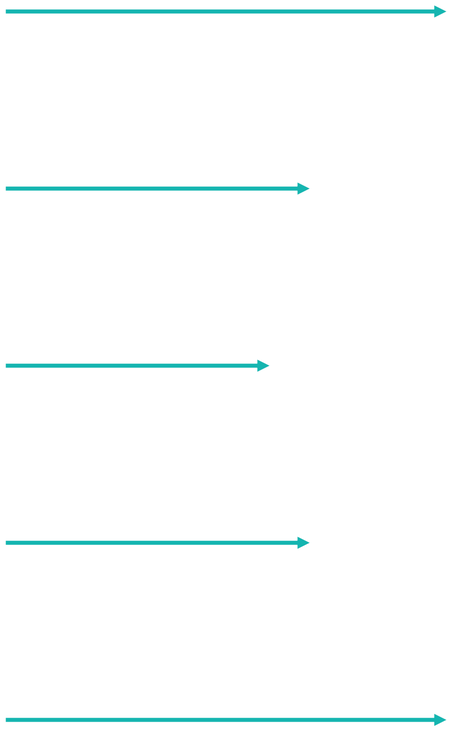
Luneburg Lens



$$\min_{\eta} \sum_{i \in \text{rays}} \left\| \mathbf{x}_{\text{target}} - \mathbf{x}_{\text{i}} \right\|^2$$

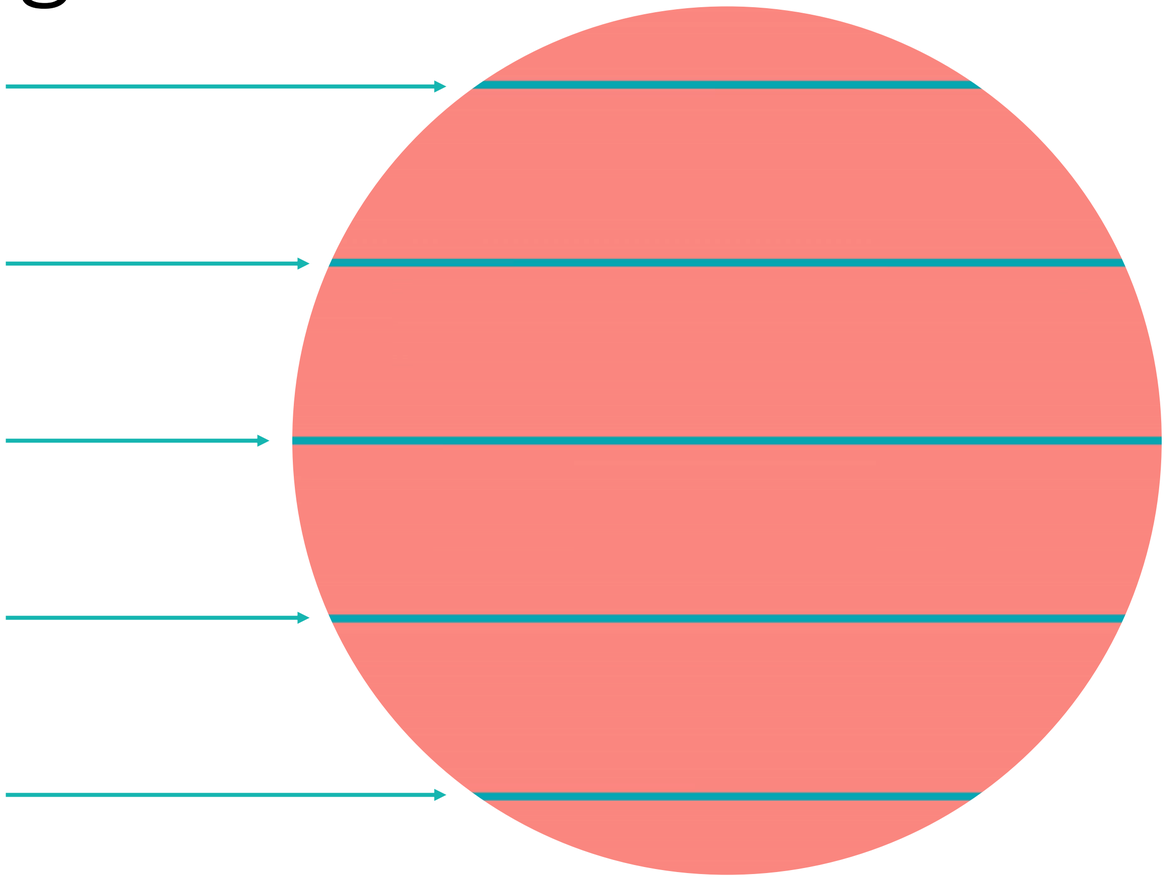
[Teh et al., 2022]

Luneburg Lens



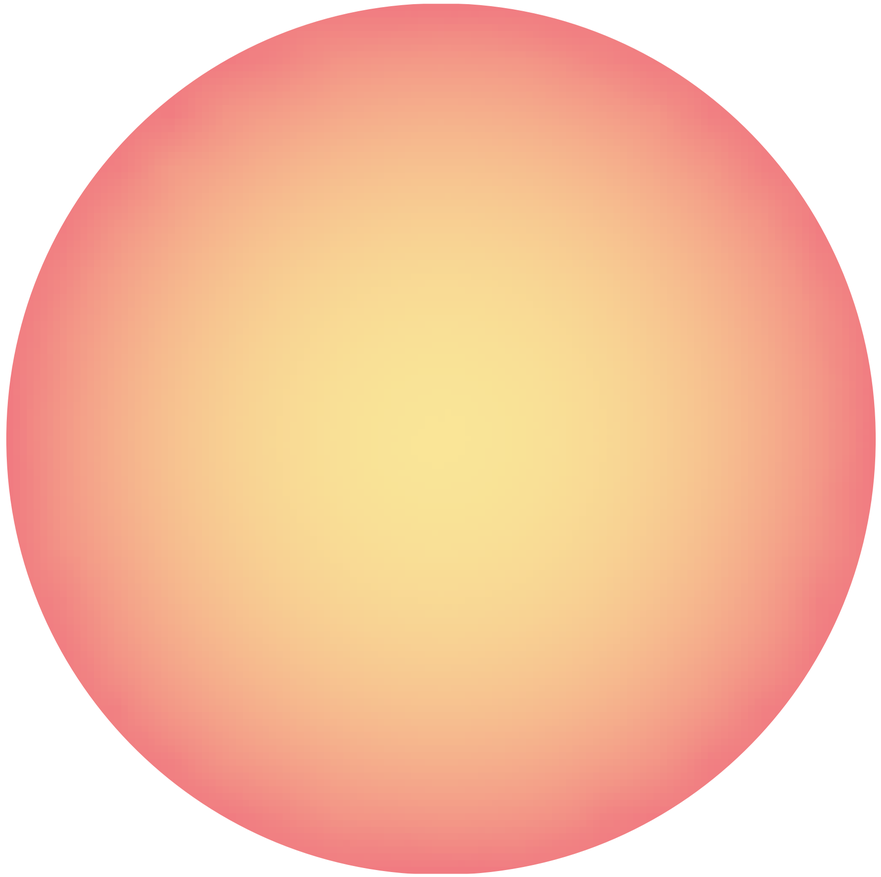
[Teh et al., 2022]

Luneburg Lens



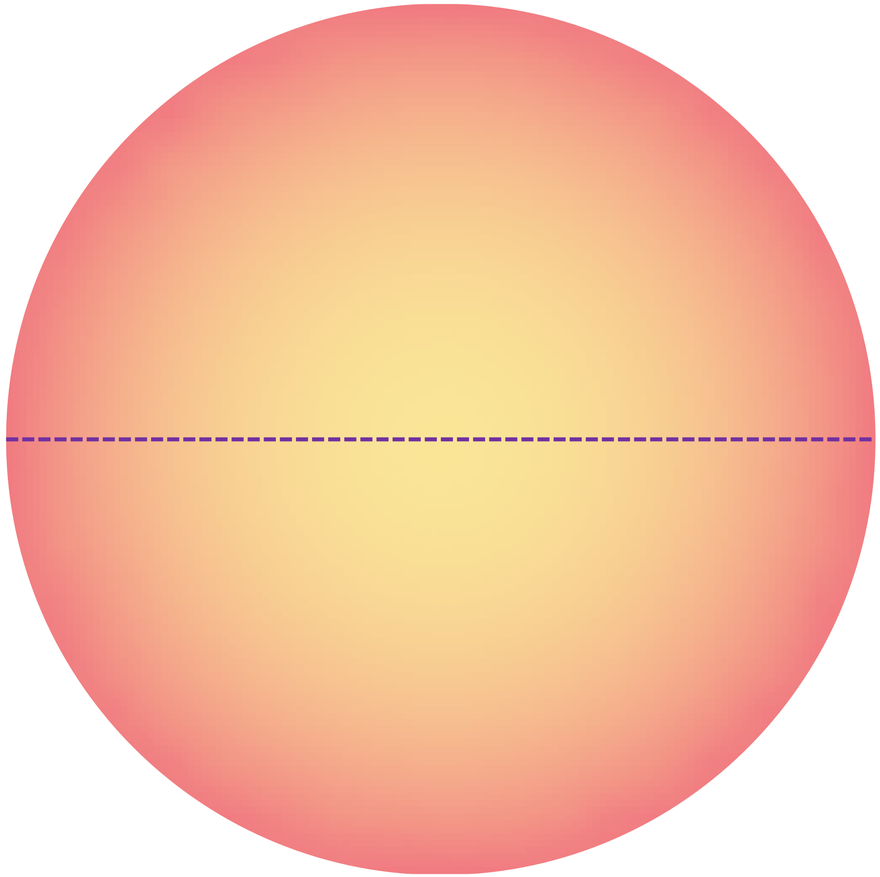
[Teh et al., 2022]

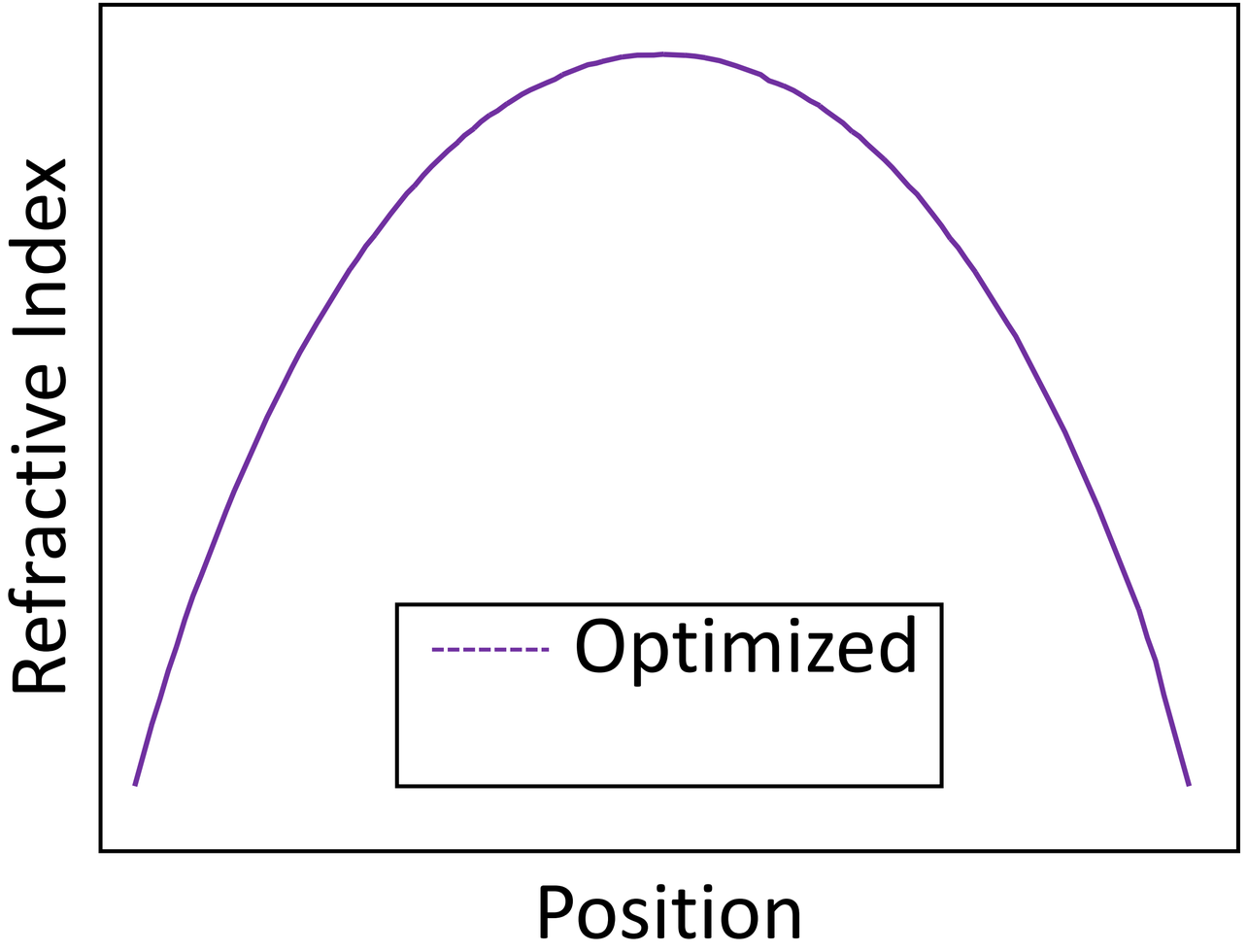
Luneburg Lens



[Teh et al., 2022]

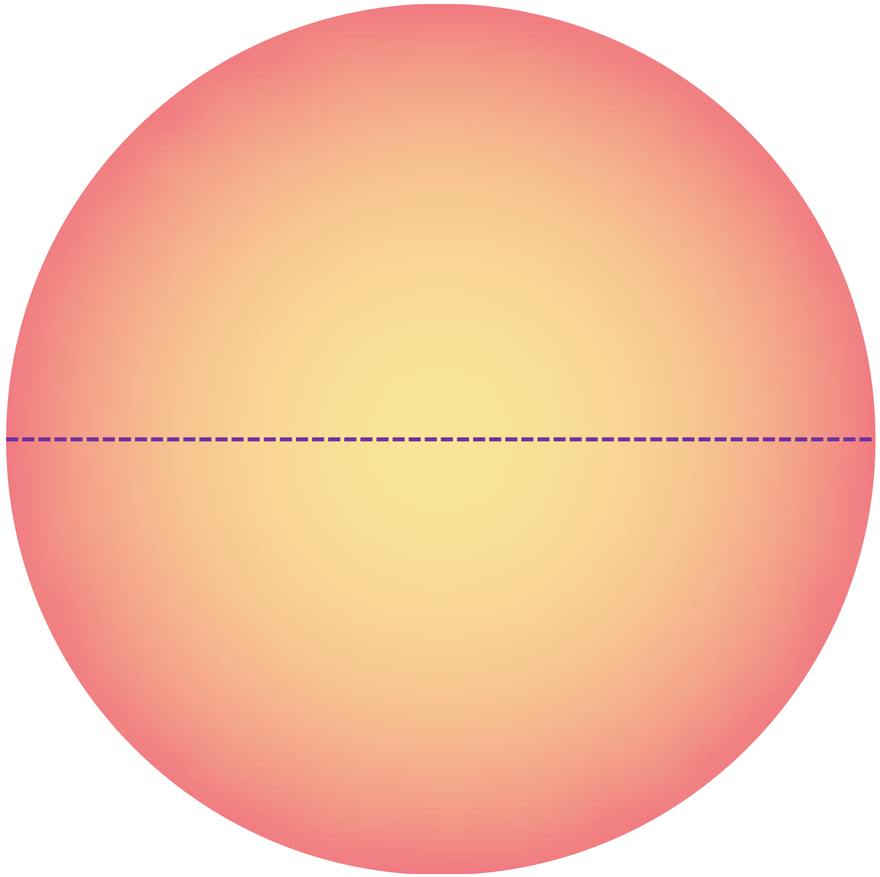
Luneburg Lens

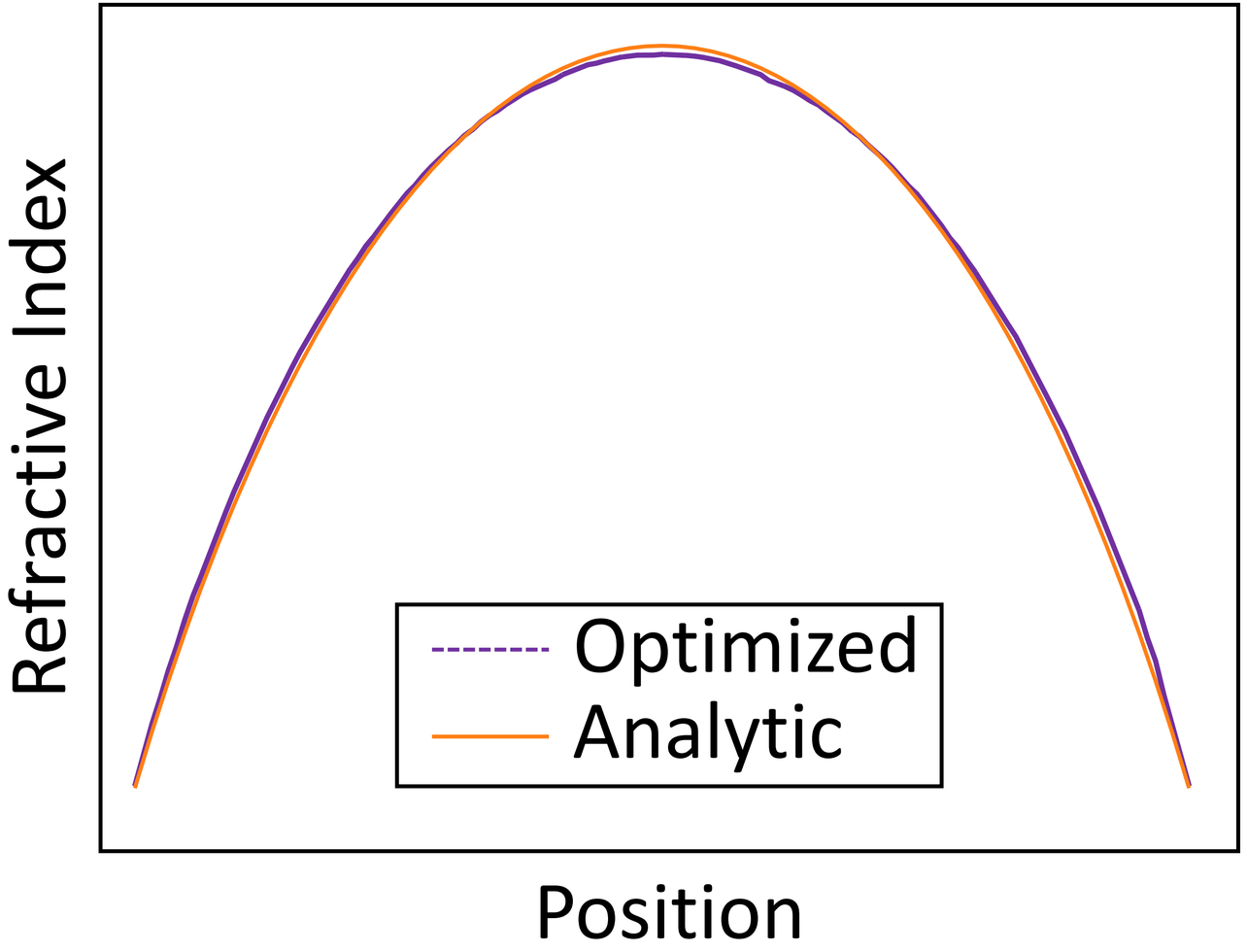




[Teh et al., 2022]

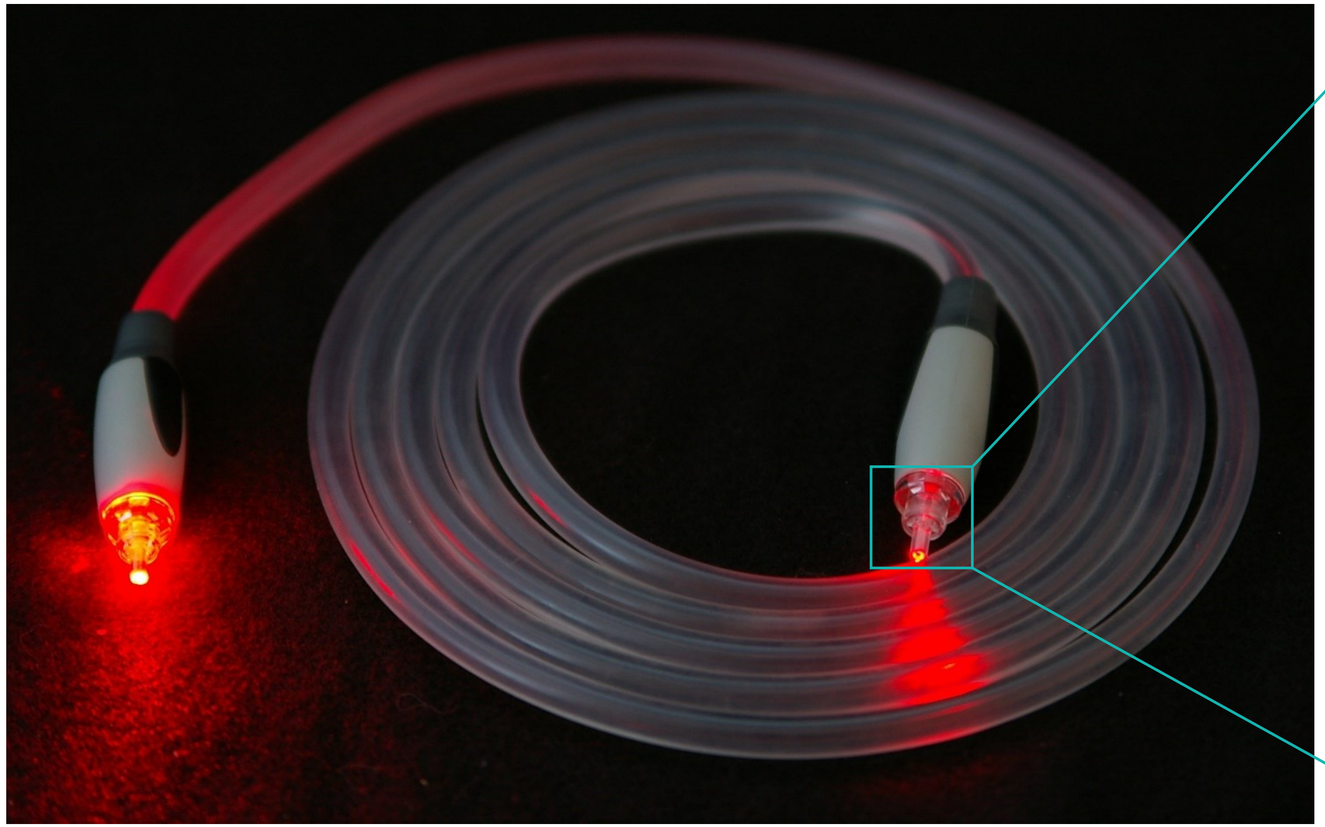
Luneburg Lens

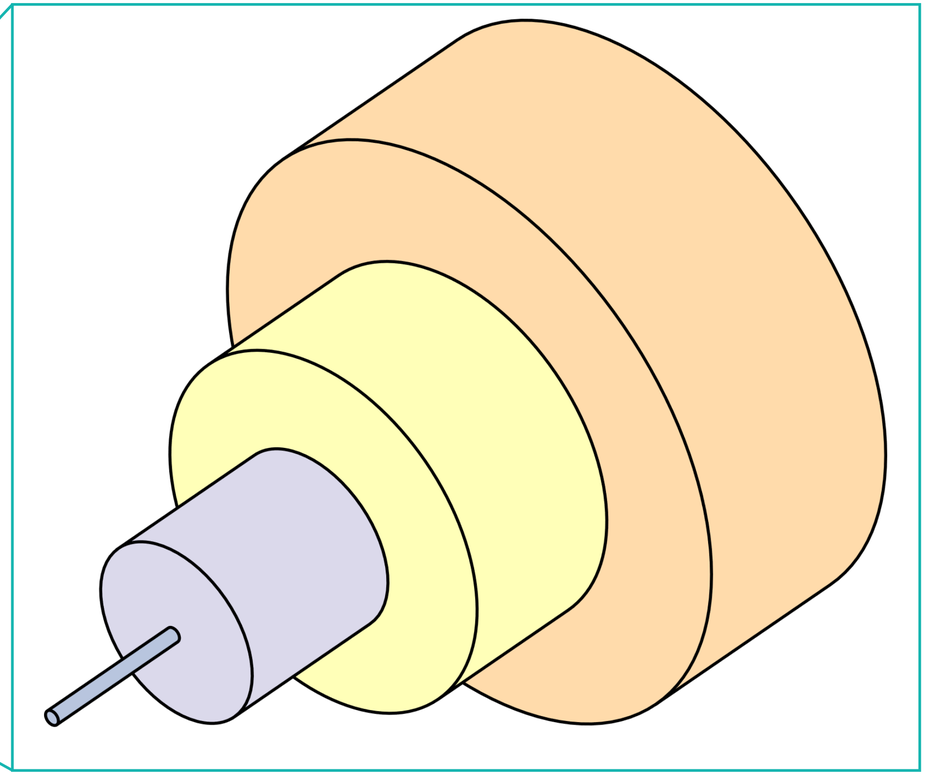




[Teh et al., 2022]

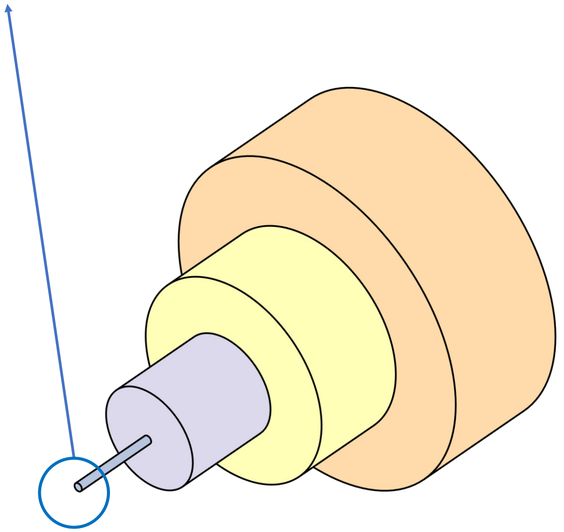
GRIN Fiber





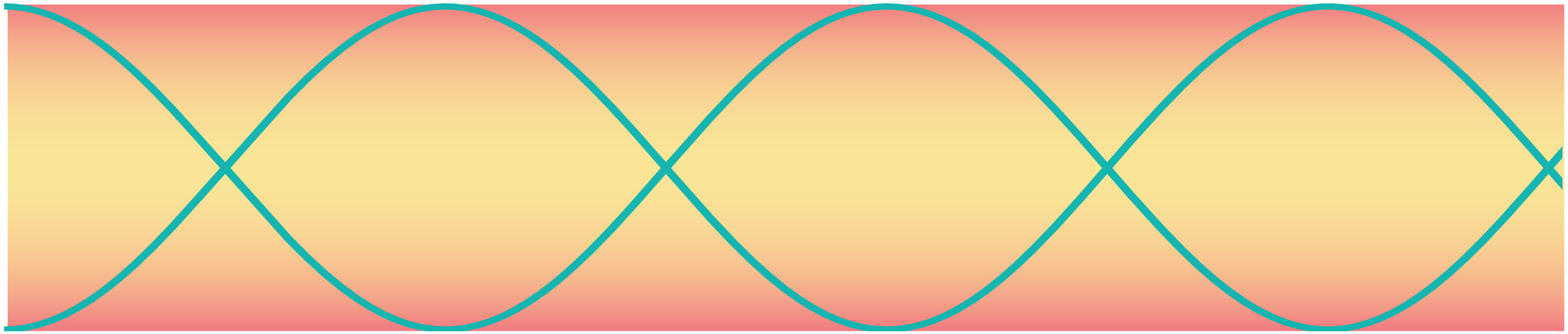
https://en.wikipedia.org/wiki/Optical_fiber

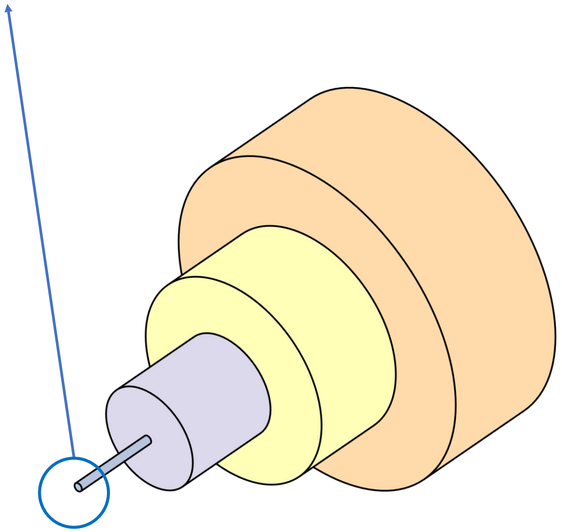
GRIN Fiber



[Teh et al., 2022]

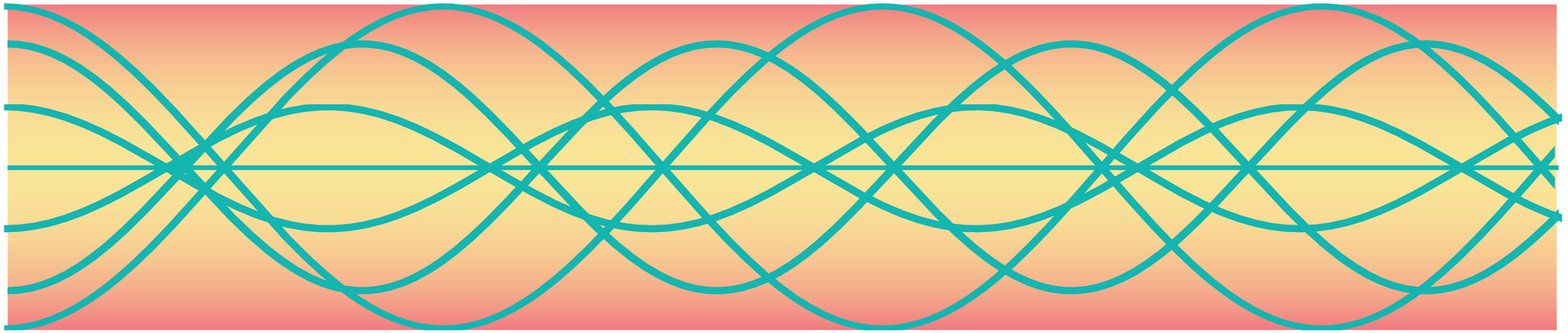
GRIN Fiber

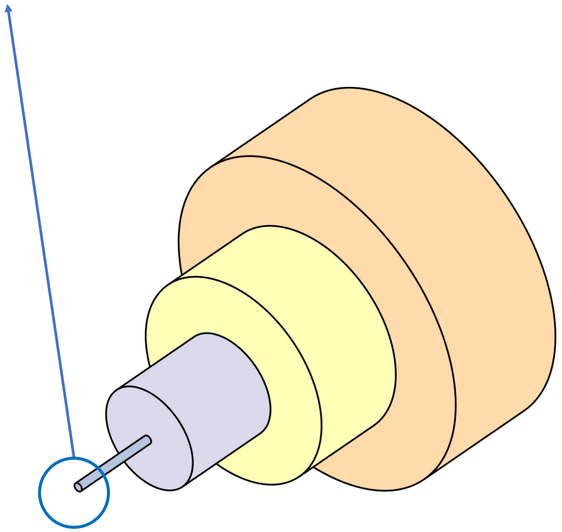




[Teh et al., 2022]

GRIN Fiber

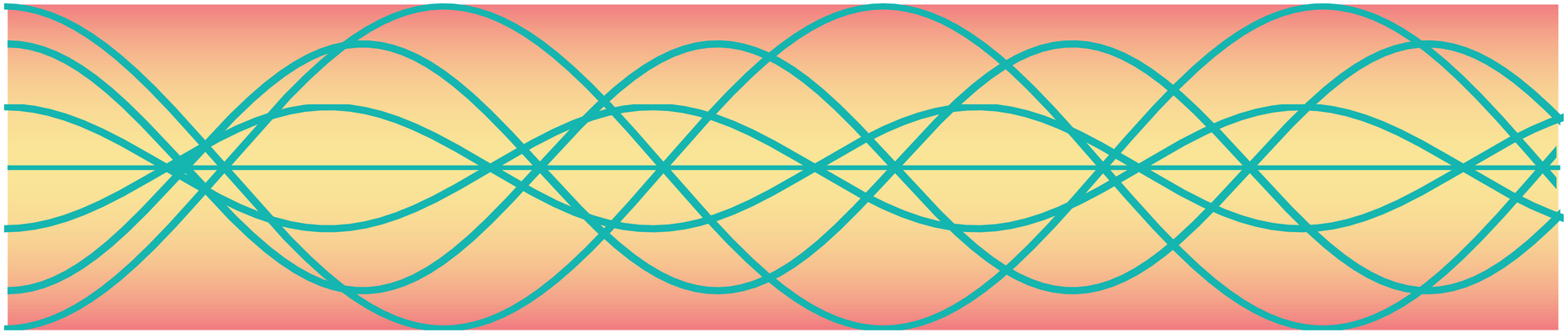


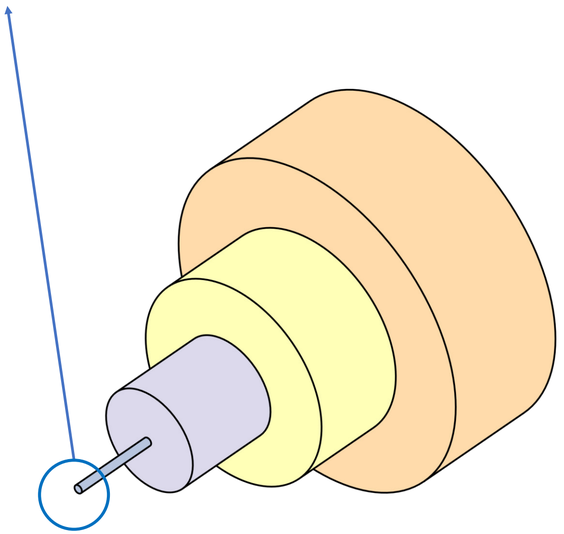


[Teh et al., 2022]

GRIN Fiber

Modal dispersion



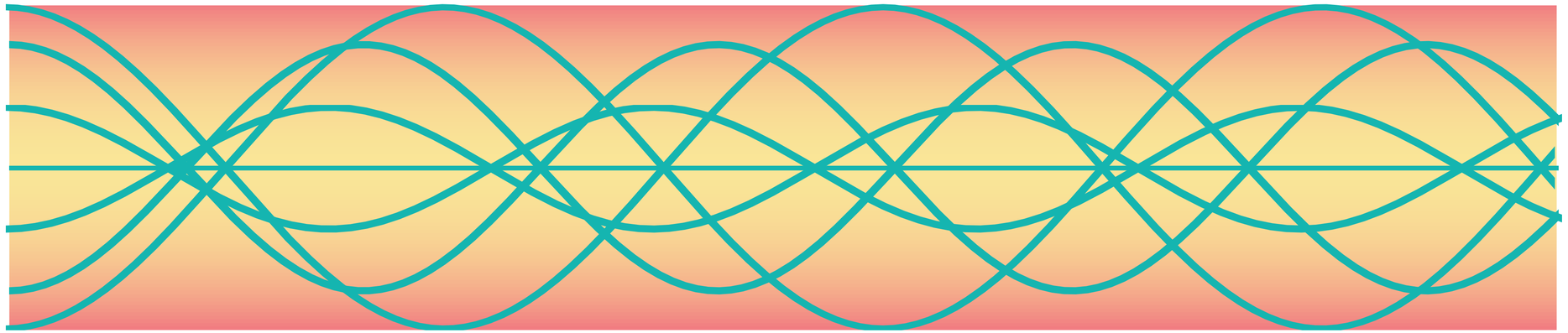


[Teh et al., 2022]

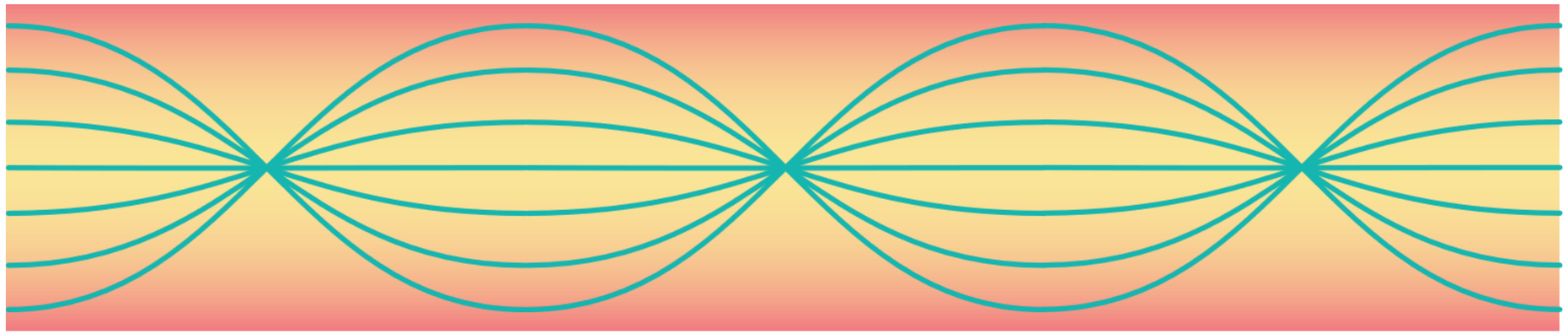
GRIN Fiber

Modal dispersion

Parabolic

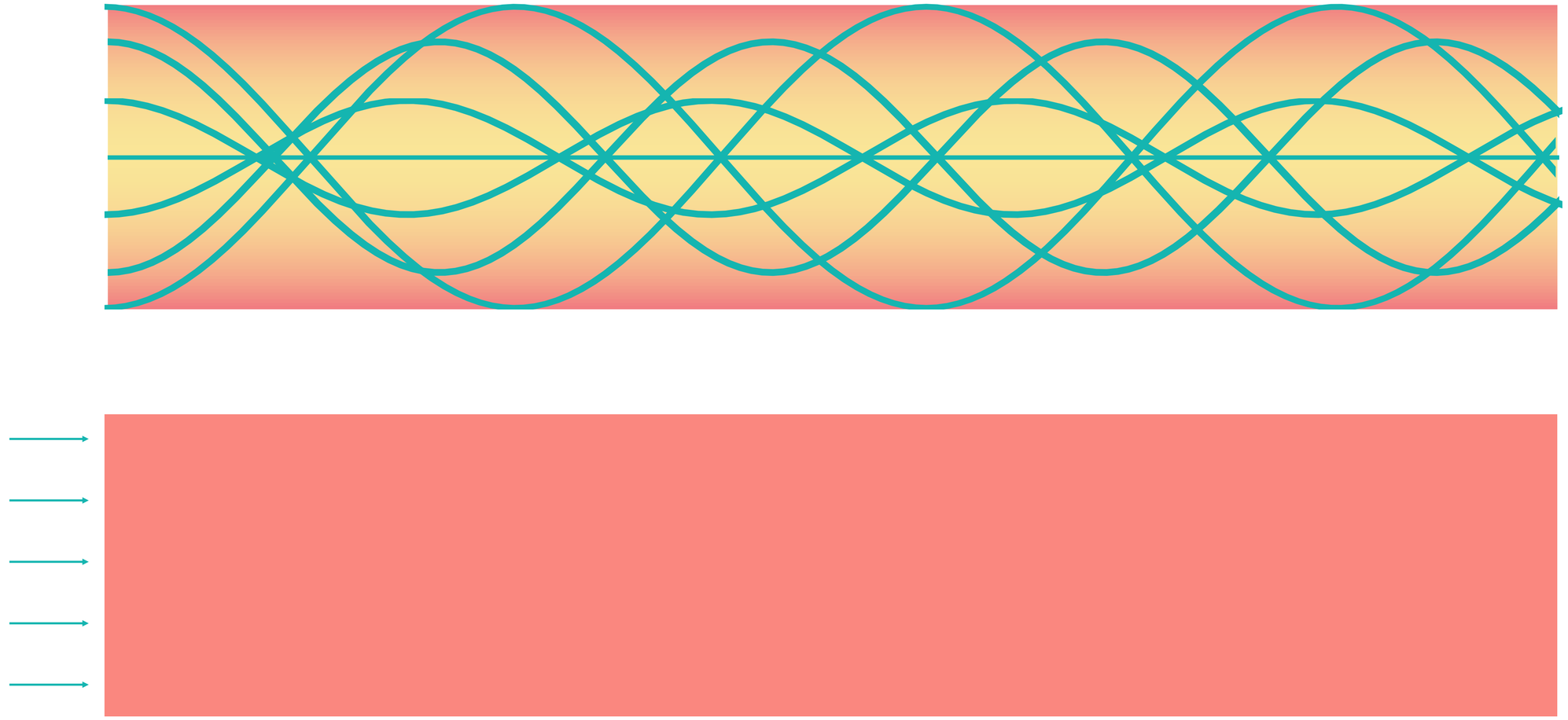


Target



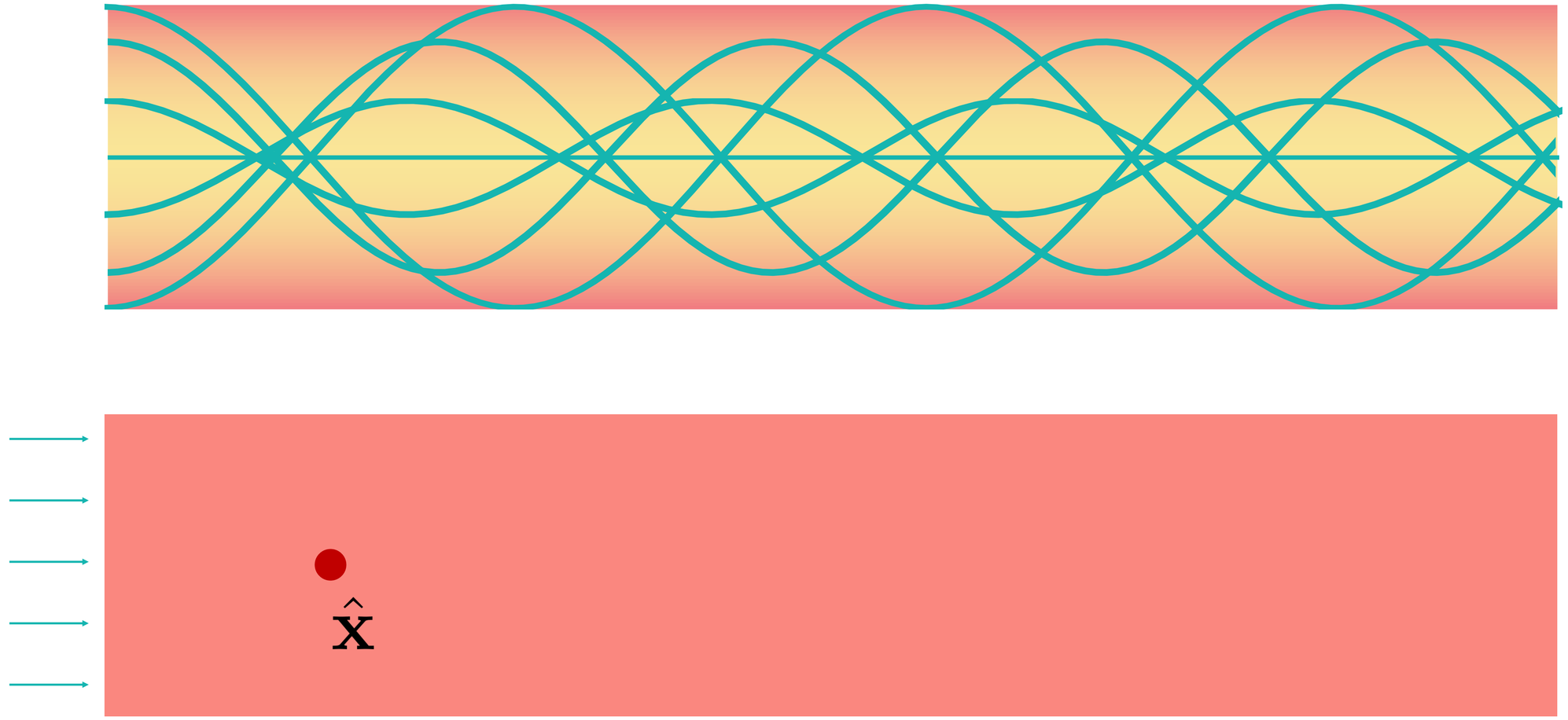
[Teh et al., 2022]

GRIN Fiber



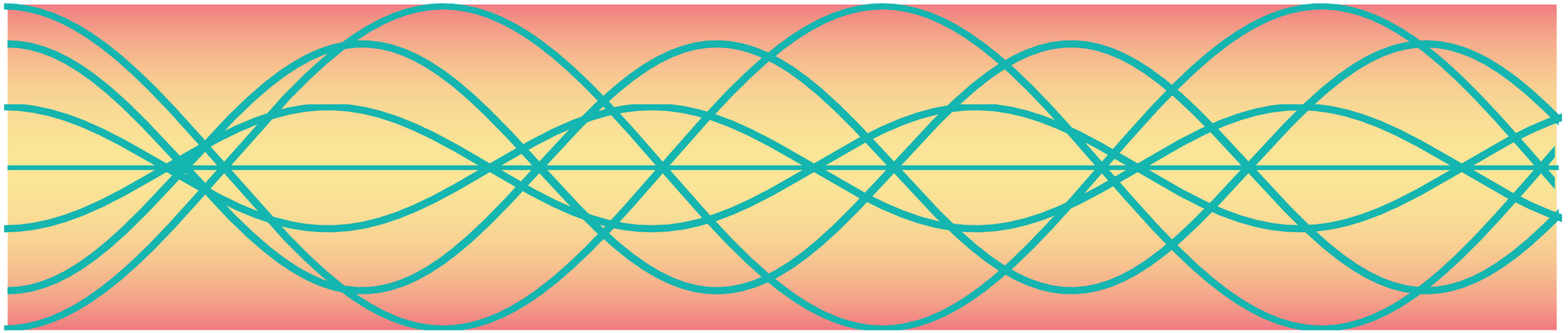
[Teh et al., 2022]

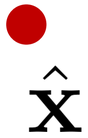
GRIN Fiber



[Teh et al., 2022]

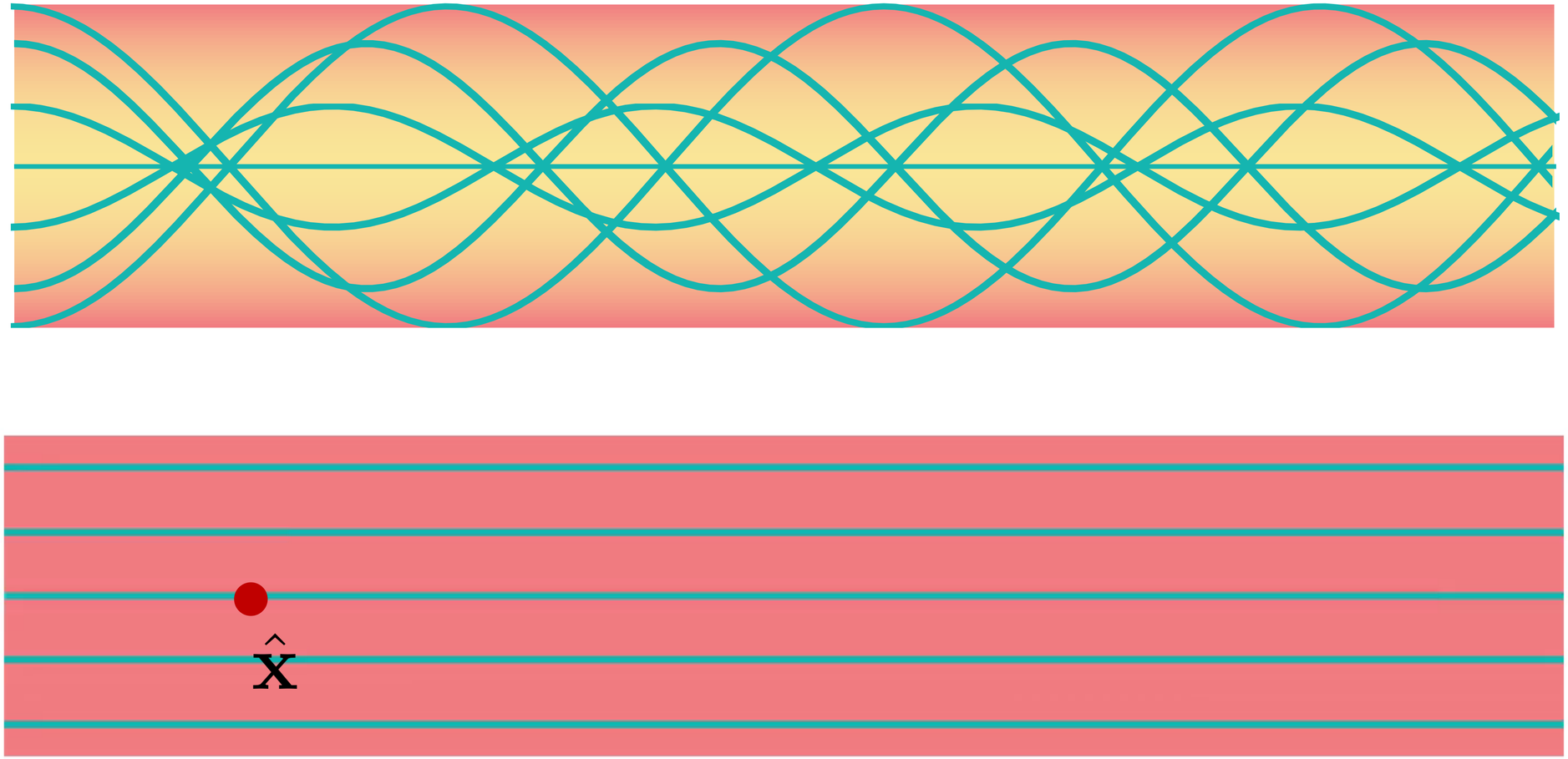
GRIN Fiber





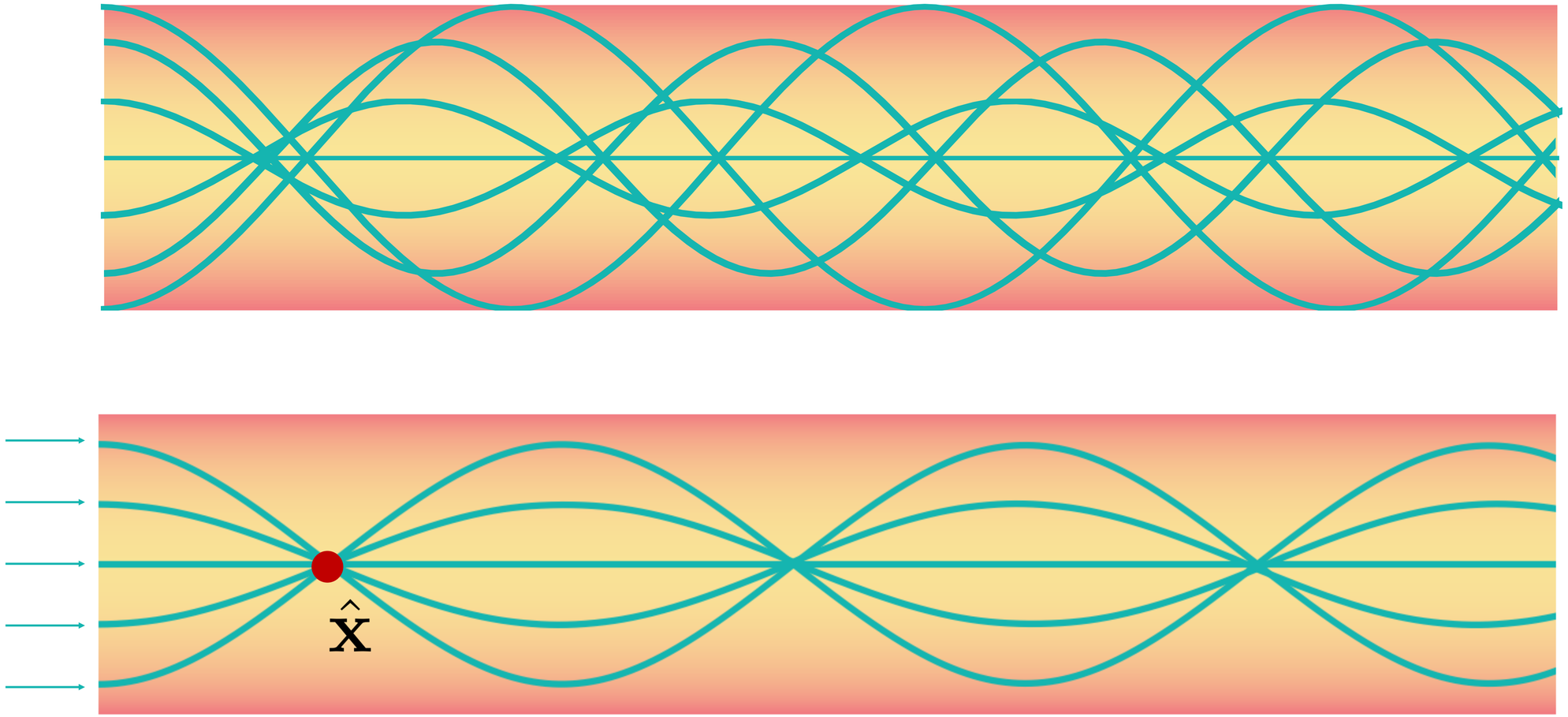
[Teh et al., 2022]

GRIN Fiber



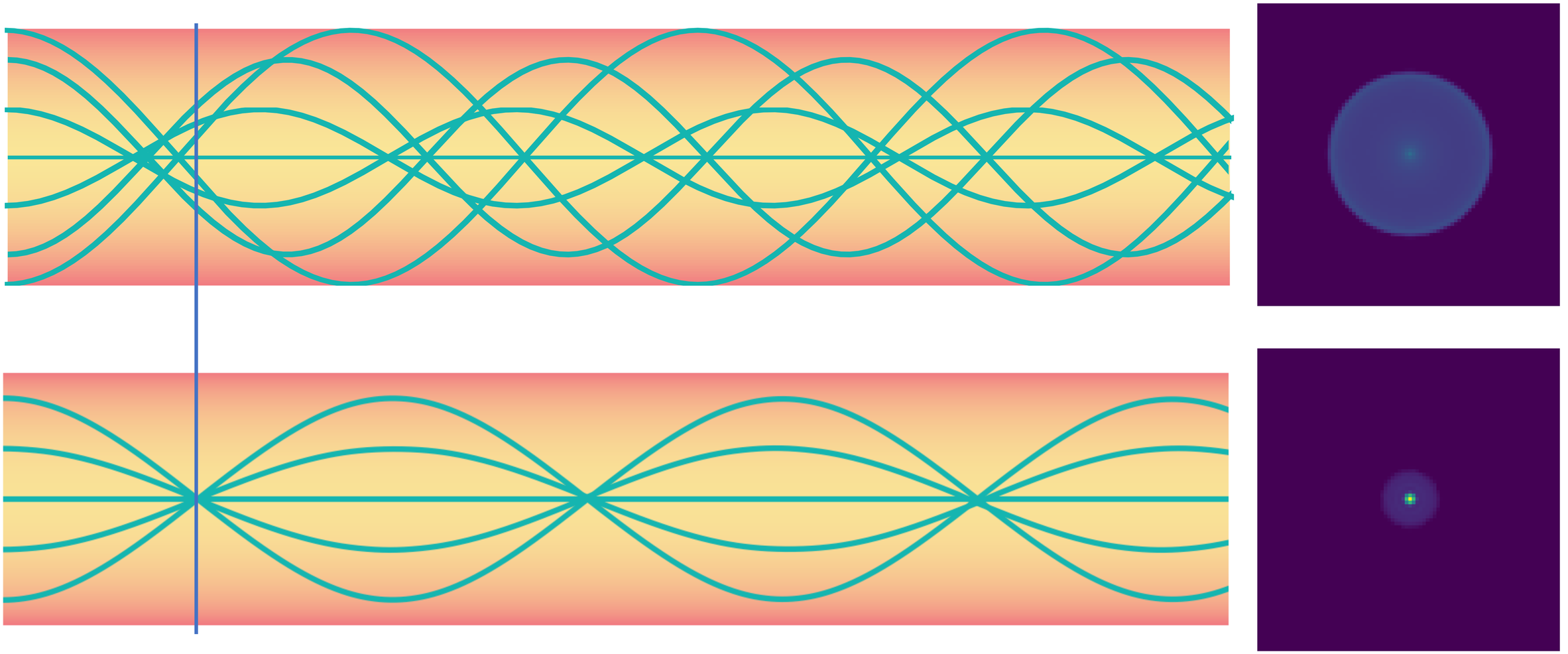
[Teh et al., 2022]

GRIN Fiber



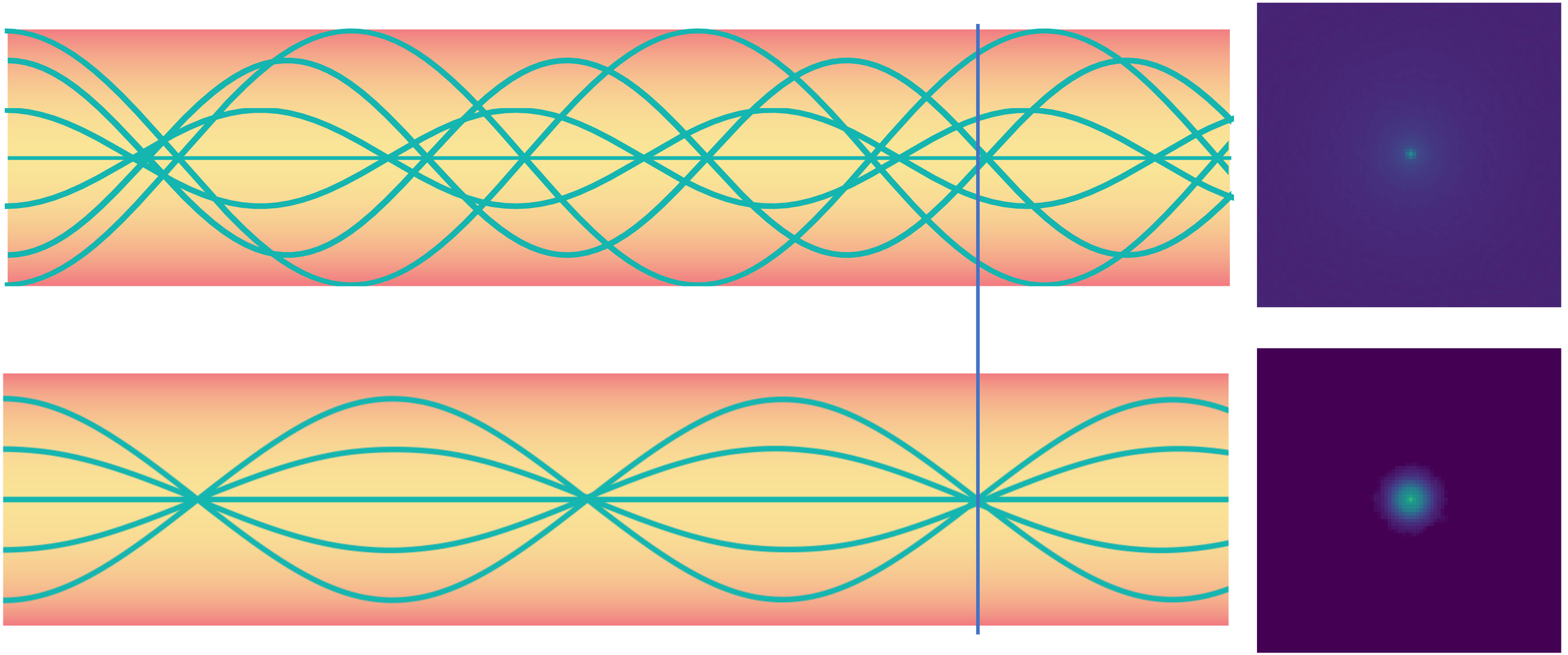
[Teh et al., 2022]

GRIN Fiber



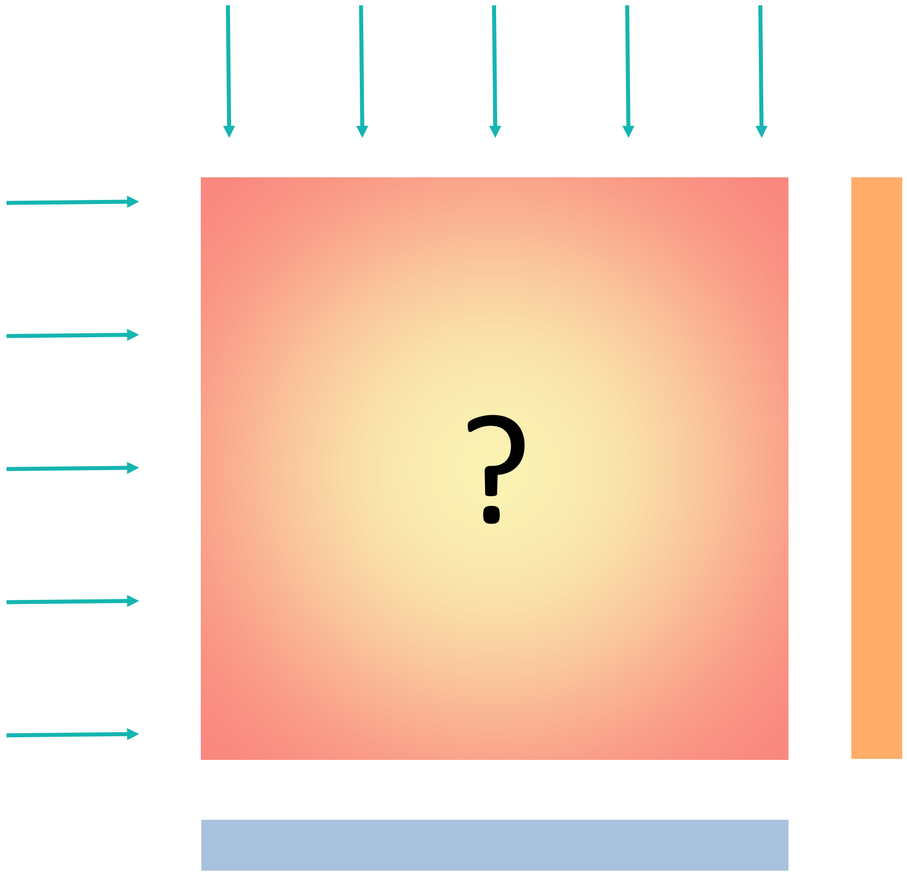
[Teh et al., 2022]

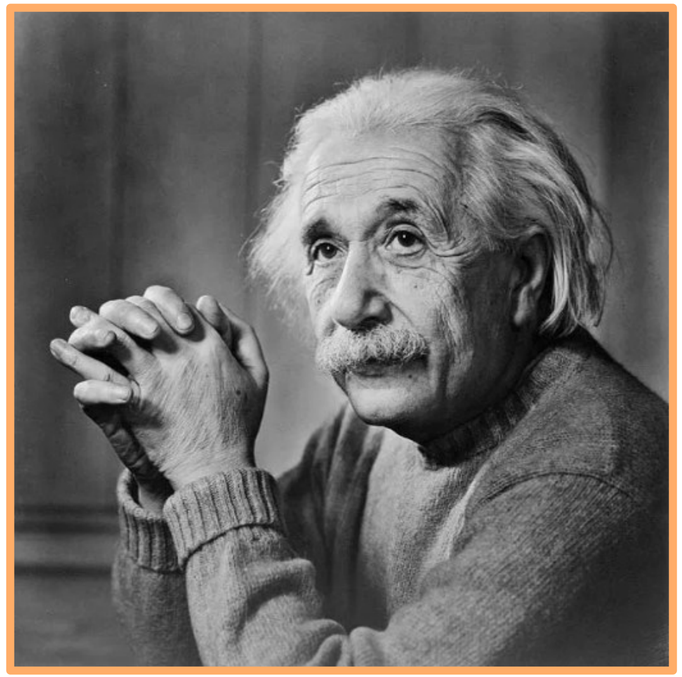
GRIN Fiber

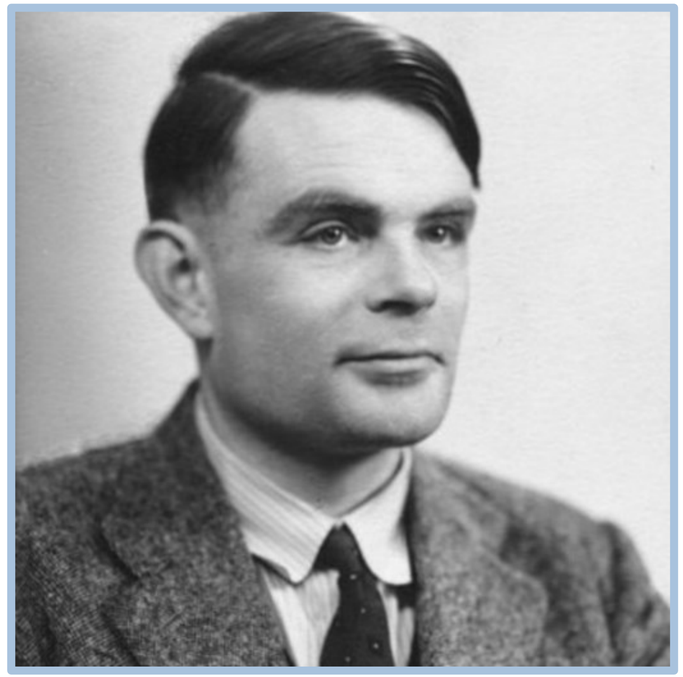


[Teh et al., 2022]

Multiview Display

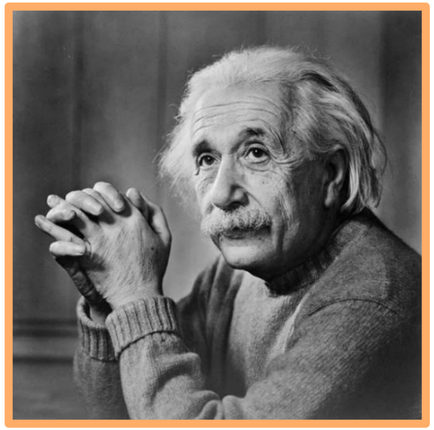




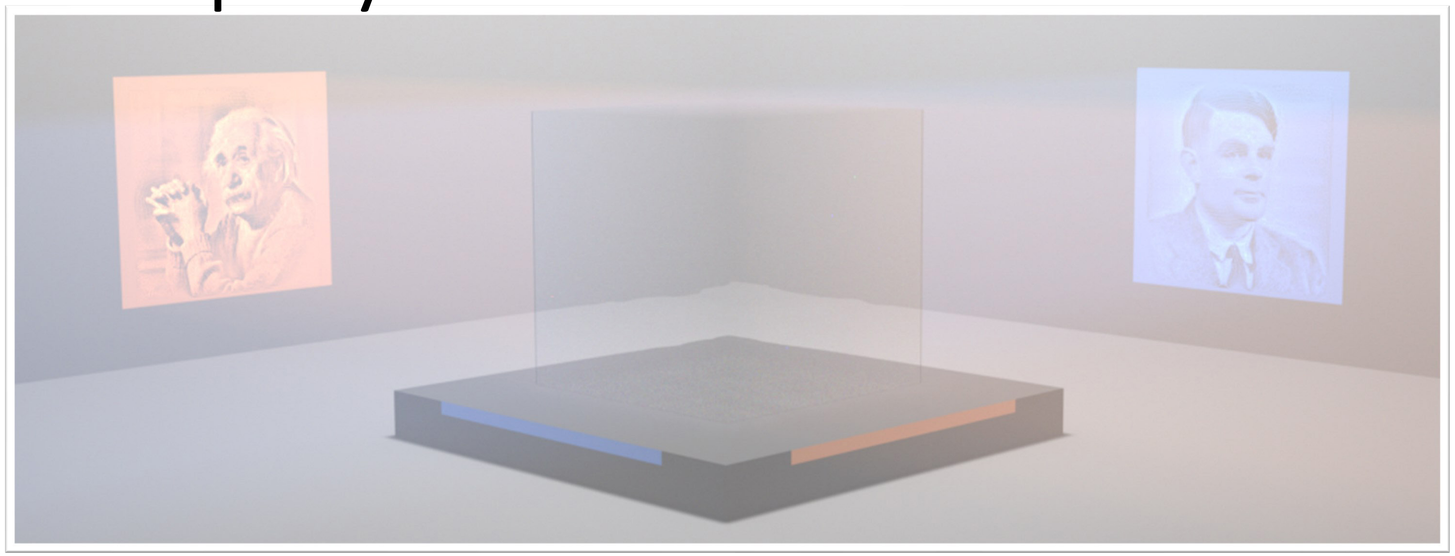


[Teh et al., 2022]

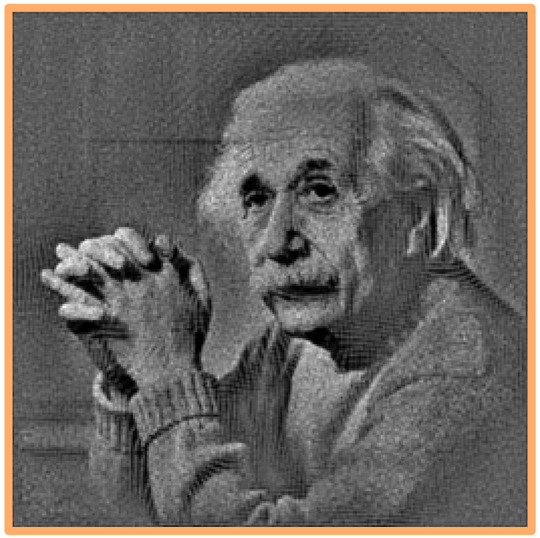
Multiview Display



Target



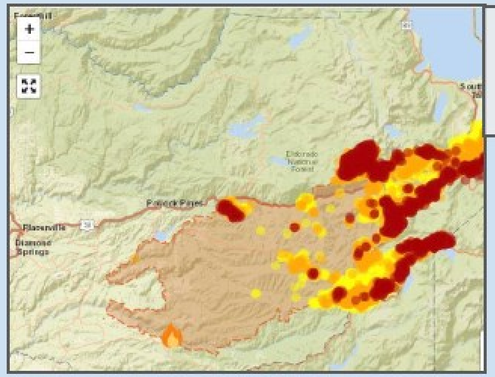
Target



optimization results

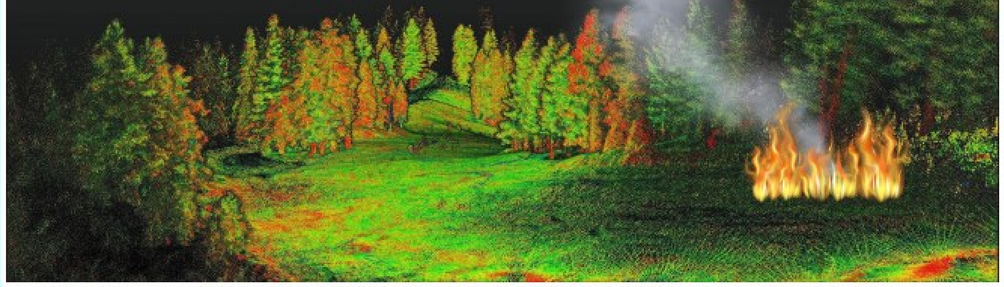
[Teh et al., 2022]

Differentiable rendering for wildfire monitoring



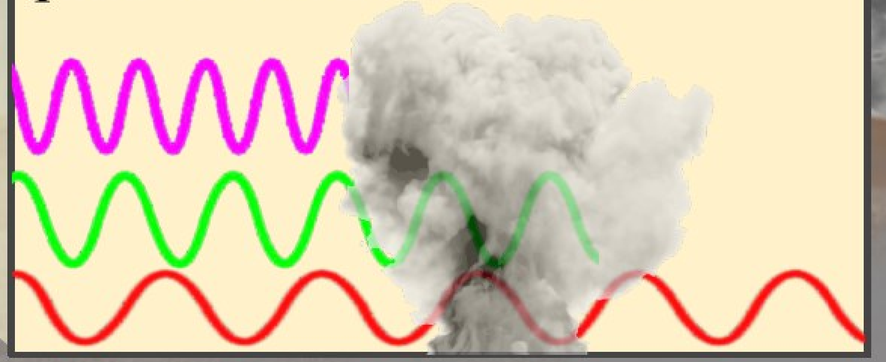
Other's product: 2D fire map, low-res

Our Product: granular 3D map of environment and fire plume



Ours: Low-flying, granular resolution; team scouts fire plume and environment up-close

Sensor wavelength trades smoke penetration and resolution



[USDA NIFA project jointly with Sebastian Scherer and Katia Sycara]

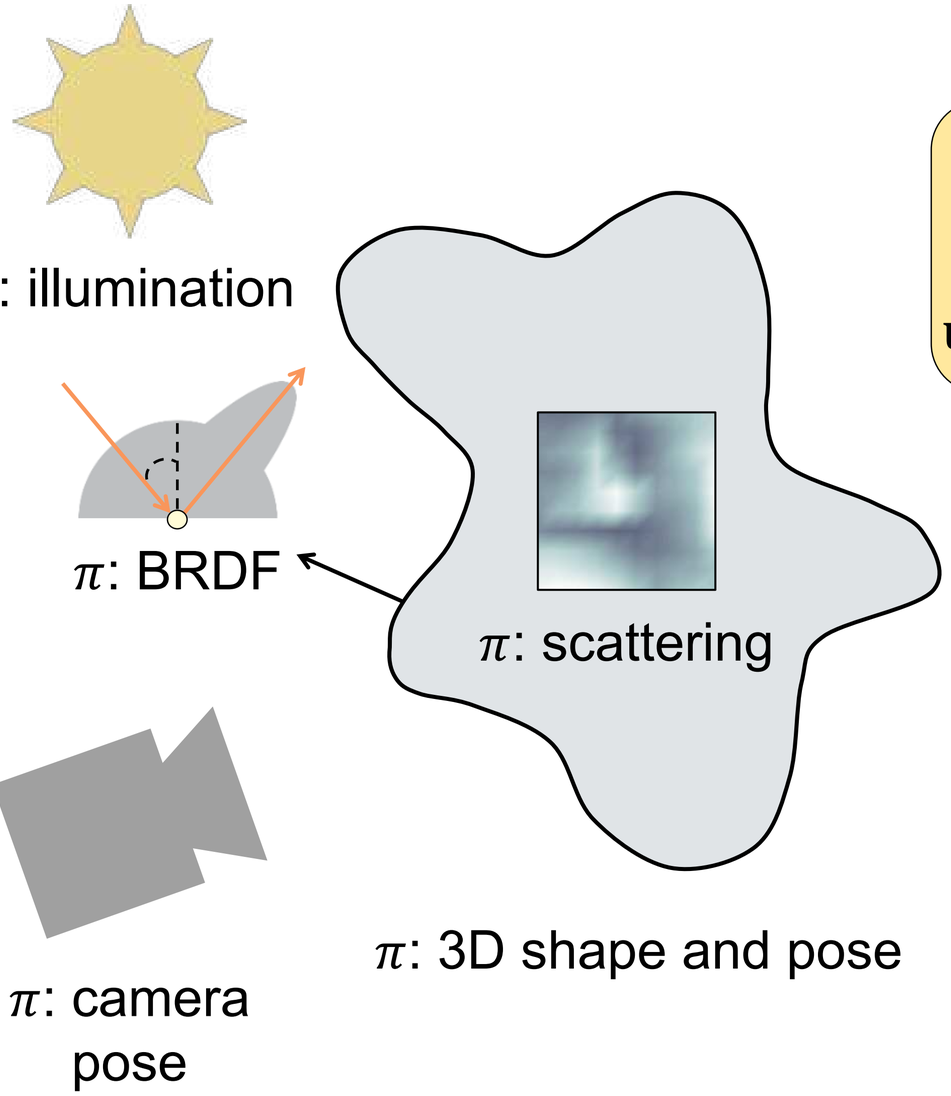
What differentiable rendering does not give us

Inverse rendering (a.k.a. analysis by synthesis)

Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{  }\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

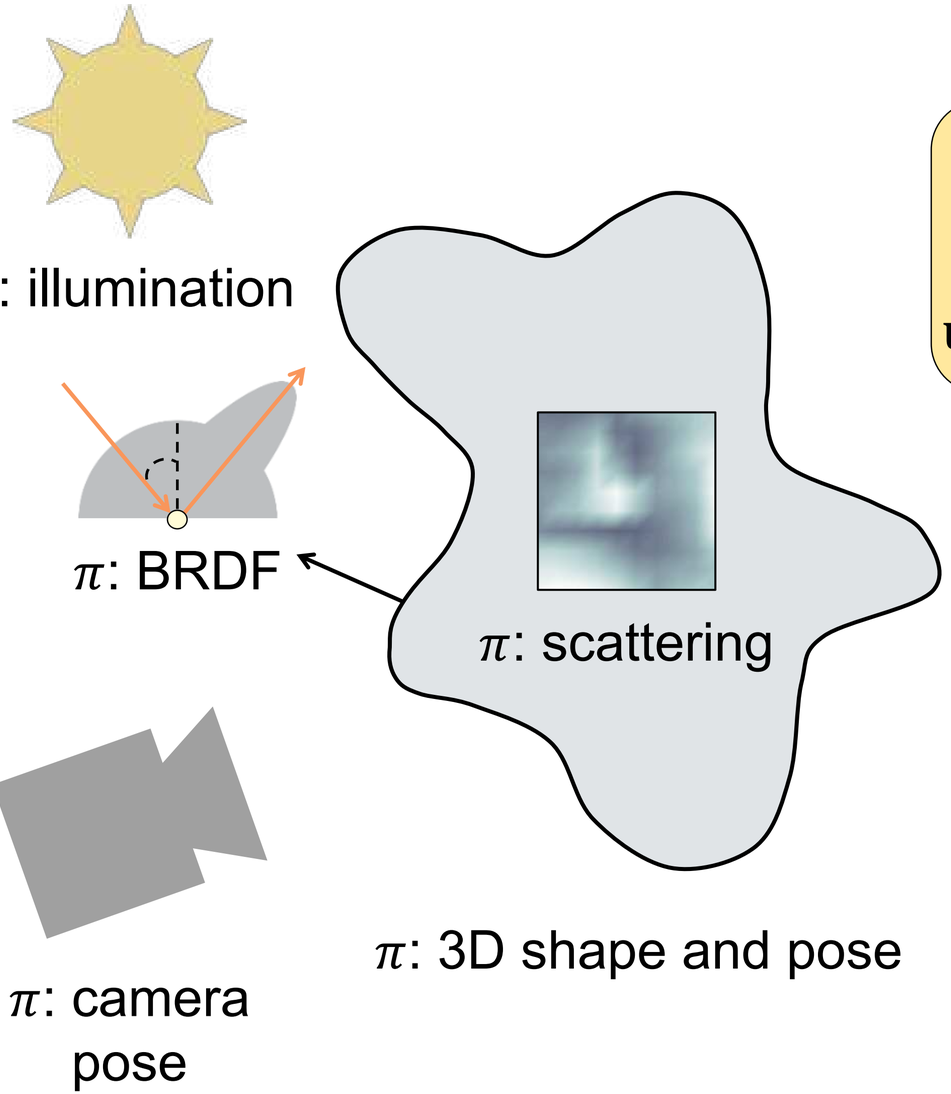
Inverse rendering (a.k.a. analysis by synthesis)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{image}\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Inverse rendering (a.k.a. analysis by synthesis)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene}\\ \text{unknowns } \pi}} \text{loss}\left[\ \text{image}\ , \text{render}\left(\begin{matrix}\text{scene}\\ \text{unknowns } \pi\end{matrix}\right)\right]$$

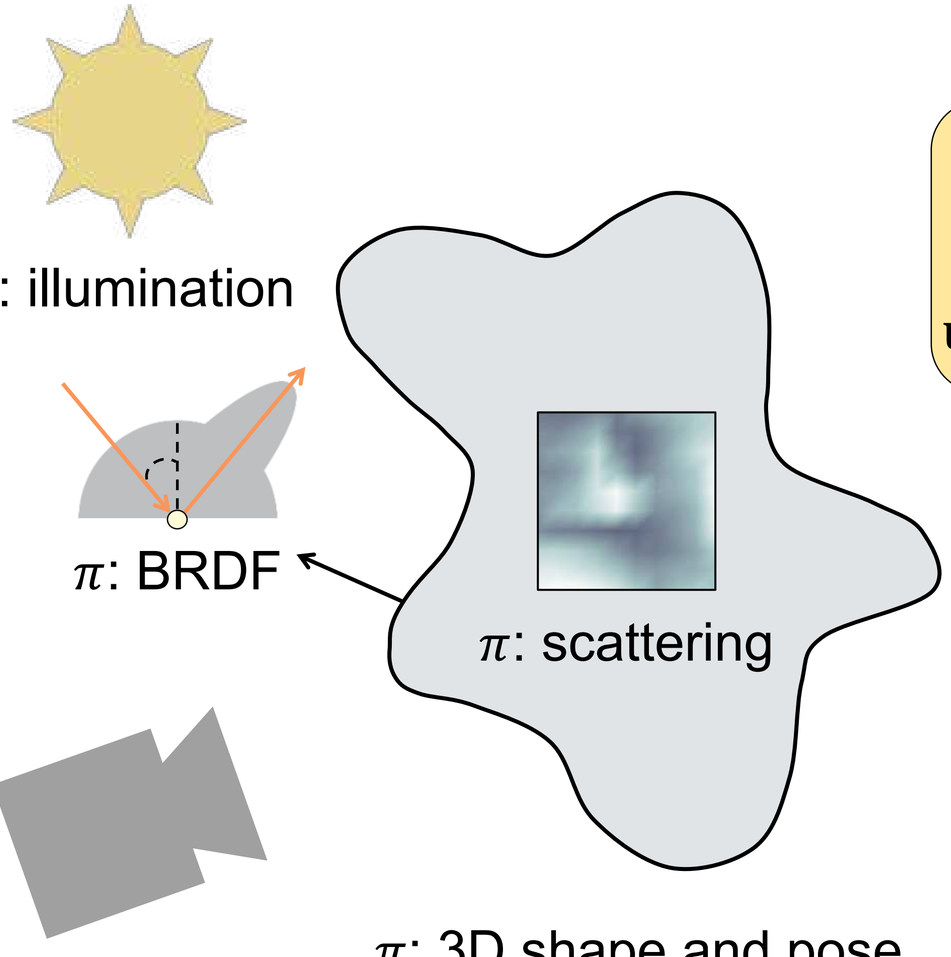
Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \frac{\text{dloss}(\pi)}{\text{d}\pi}$

Inverse rendering (a.k.a. analysis by synthesis)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\text{  }, \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Stochastic gradient descent (e.g., Adam):

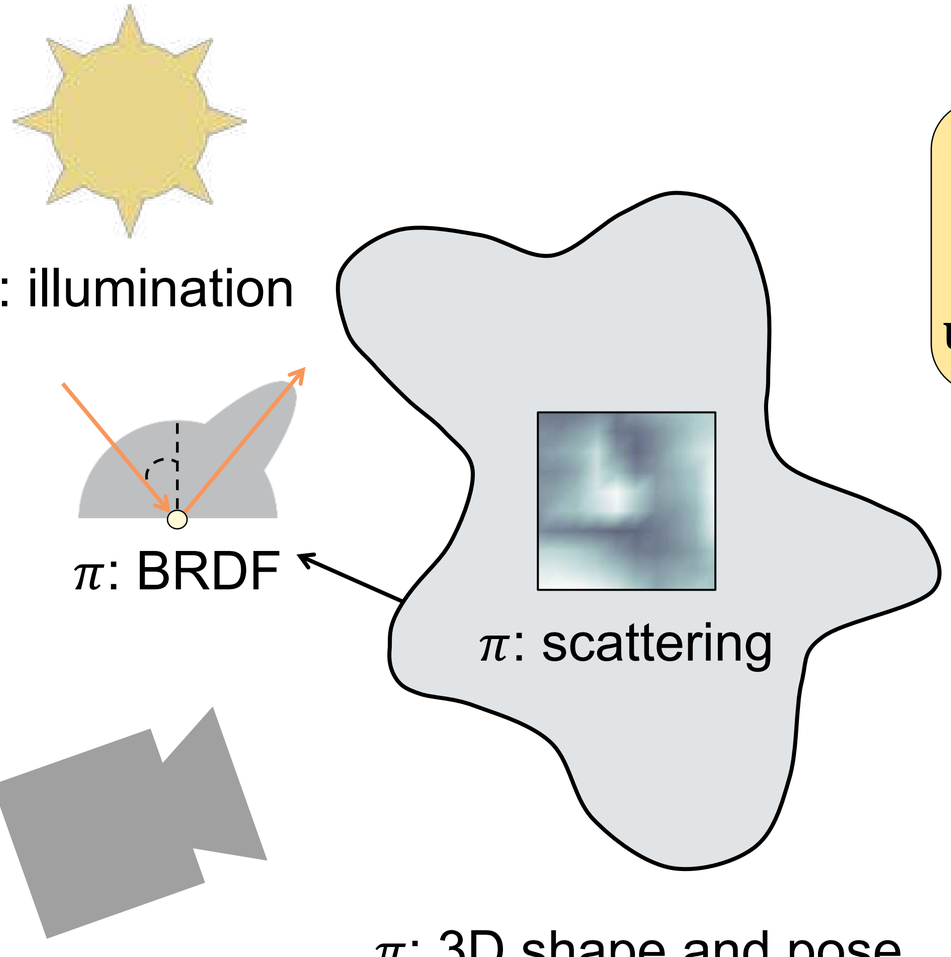
initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \frac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

Inverse rendering (a.k.a. analysis by synthesis)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\text{  }, \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \frac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

Why we need good initializations

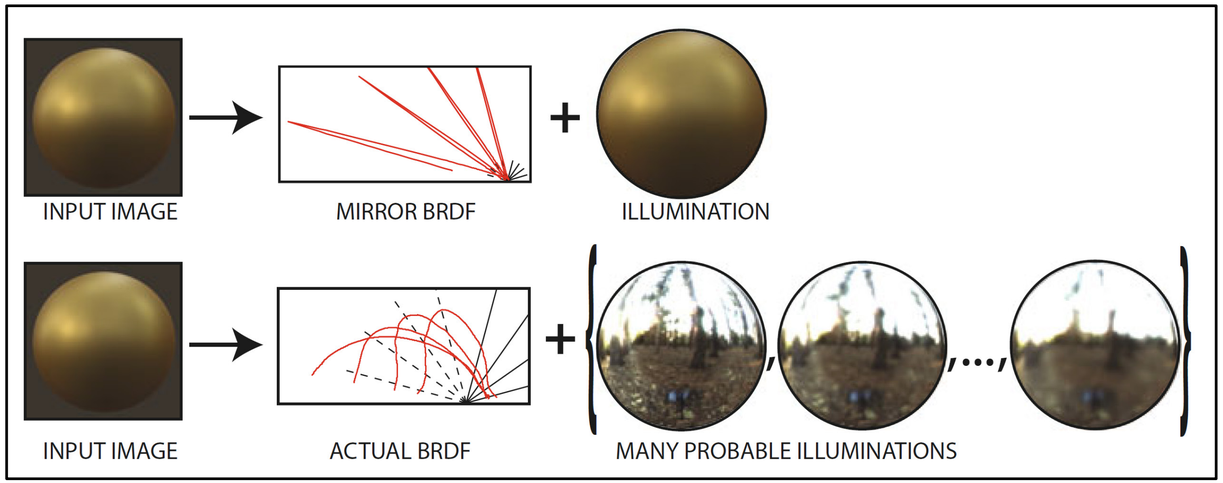
- Analysis-by-synthesis objectives are highly non-convex, non-linear
  - Multiple *local* minima

Why we need good initializations

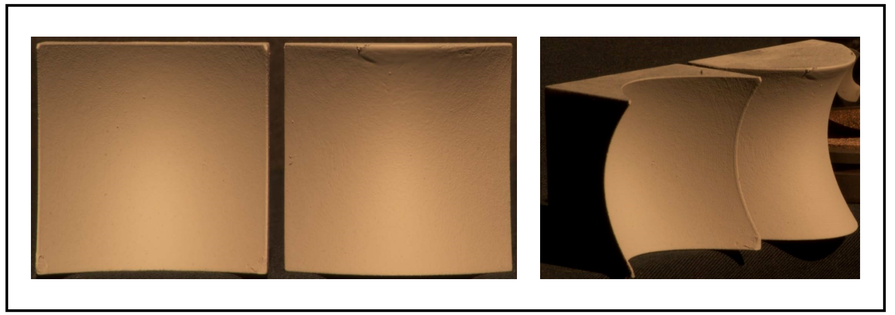
- Analysis-by-synthesis objectives are highly non-convex, non-linear
  - Multiple *local* minima
- Ambiguities exist between different parameters
  - Multiple *global* minima

Why we need good initializations

- Analysis-by-synthesis objectives are highly non-convex, non-linear
  - Multiple *local* minima
- Ambiguities exist between different parameters
  - Multiple *global* minima



Ambiguities between BRDF and lighting [Romeiro and Zickler 2010]



Ambiguities between shape and lighting [Xiong et al. 2015]

Ambiguities between scattering parameters [Zhao et al. 2014]

Inverse rendering (a.k.a. analysis by synthesis)

Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{  }\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

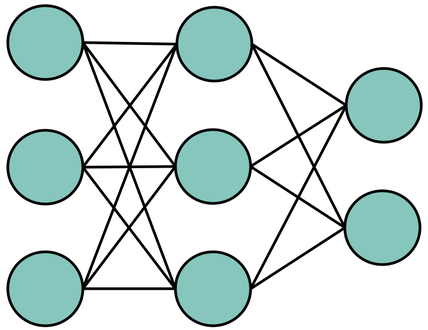
update $\pi \leftarrow \pi + \eta \cdot \frac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

Inverse rendering (a.k.a. analysis by synthesis)

Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{  }\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$



Neural network

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \dfrac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

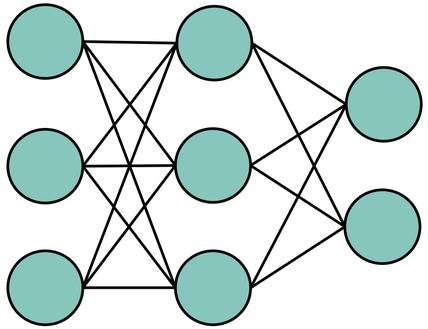
Inverse rendering (a.k.a. analysis by synthesis)

Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene}\\ \text{unknowns } \pi}} \text{loss}\left[\ \text{  }\ , \text{render}\left(\begin{matrix}\text{scene}\\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Learned initializations help:

- avoid local minima
- accelerate convergence



Neural network

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \dfrac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

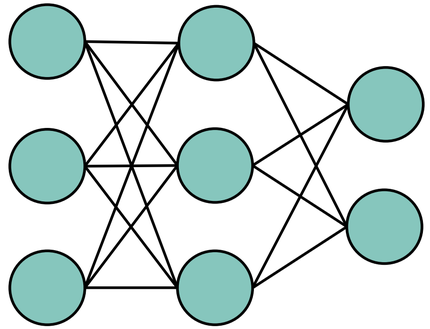
Inverse rendering (a.k.a. analysis by synthesis)

Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene}\\ \text{unknowns } \pi}} \text{loss}\left[\ \text{image}\ , \text{render}\left(\begin{matrix}\text{scene}\\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Learned initializations help:

- avoid local minima
- accelerate convergence



Neural network

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

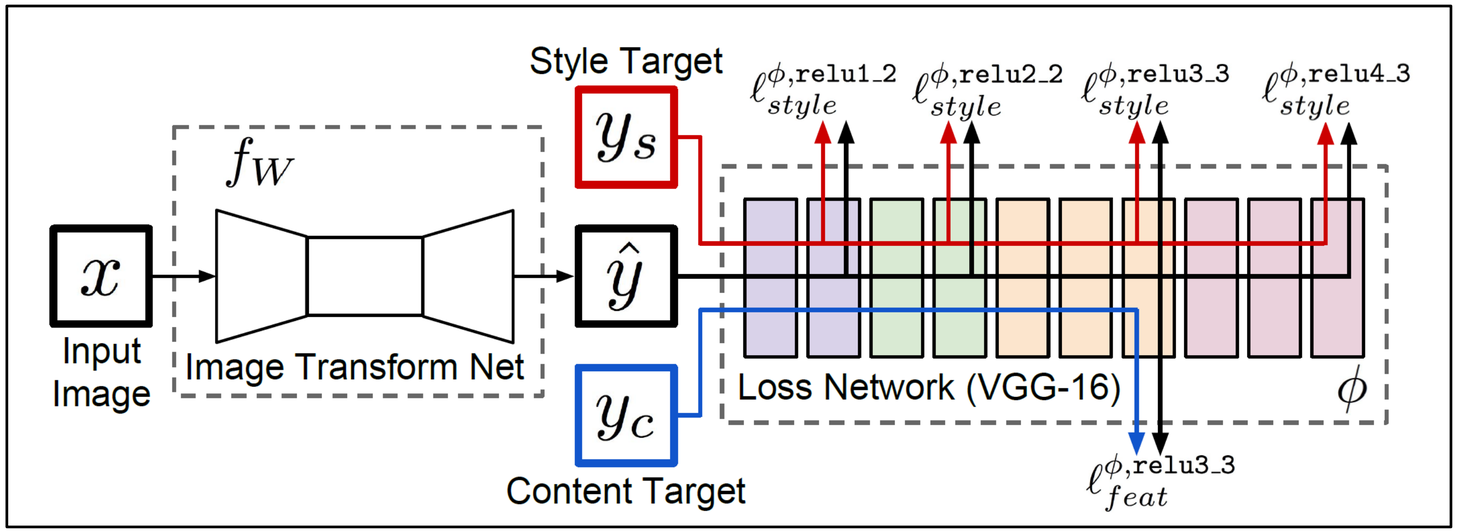
while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \dfrac{\text{dloss}(\pi)}{\text{d}\pi}$

Differentiable rendering

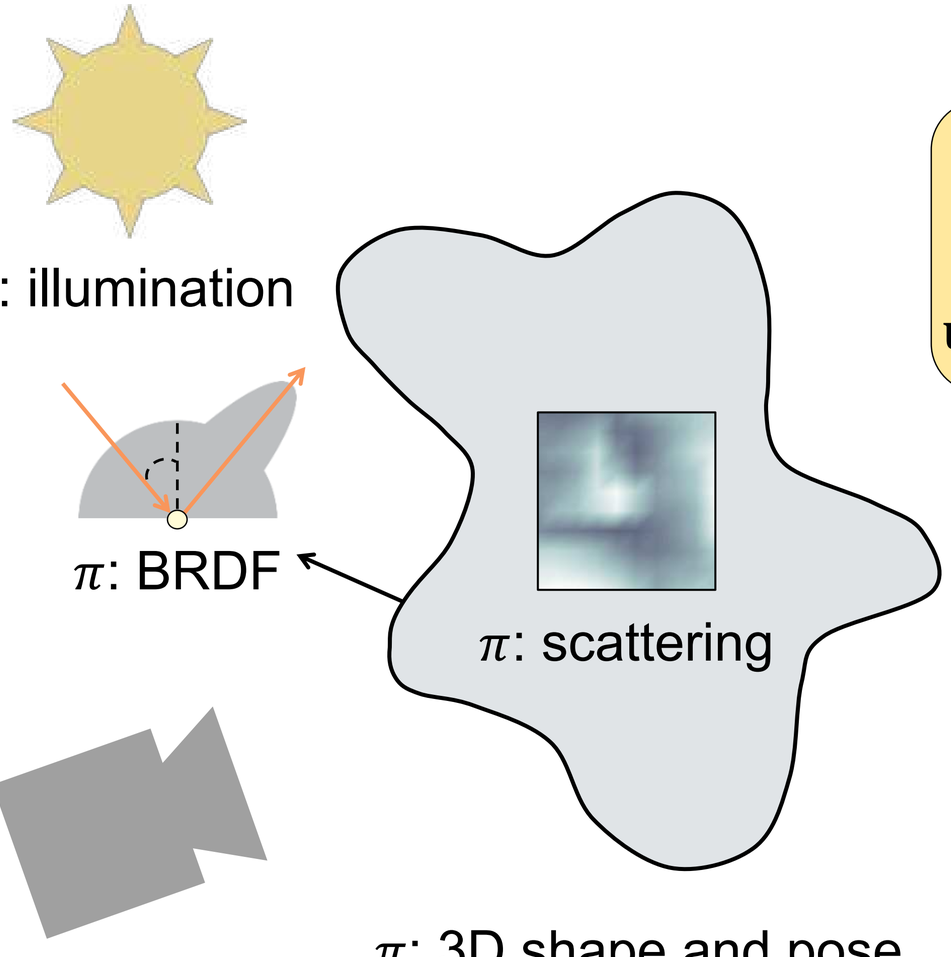
Why we need discriminative loss functions

- Well-designed loss functions can help reduce ambiguities
- Perceptual losses can help emphasize design aspects that matter
- Differentiable rendering can be combined with any loss function that can be backpropagated through



VGG-based *perceptual loss* [Johnson et al. 2016]

Inverse rendering (a.k.a. analysis by synthesis)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\text{  }, \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Stochastic gradient descent (e.g., Adam):

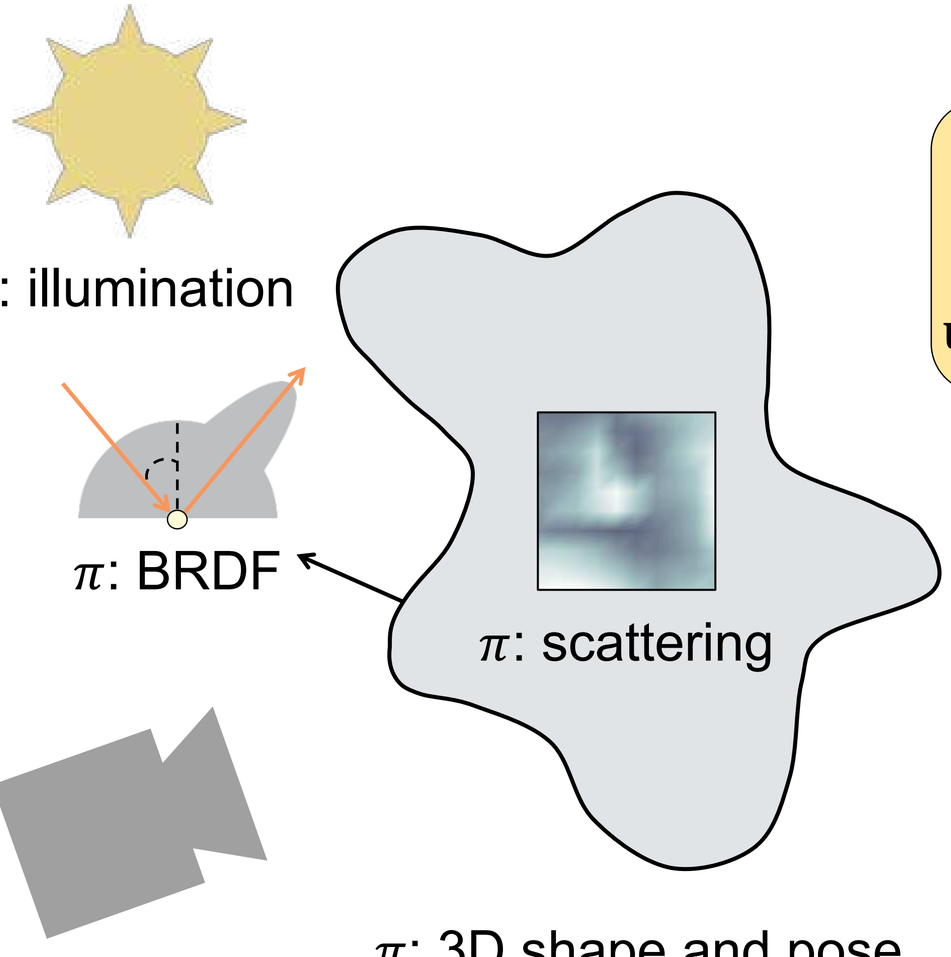
initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \frac{\mathrm{dloss}(\pi)}{\mathrm{d}\pi}$

Differentiable rendering

Inverse rendering (a.k.a. analysis by synthesis)



Analysis-by-synthesis optimization:

$$\min_{\substack{\text{scene} \\ \text{unknowns } \pi}} \text{loss}\left[\ \text{  }\ , \text{render}\left(\begin{matrix}\text{scene} \\ \text{unknowns } \pi\end{matrix}\right)\right]$$

Stochastic gradient descent (e.g., Adam):

initialize $\pi \leftarrow \pi_0$

while (not converged)

update $\pi \leftarrow \pi + \eta \cdot \frac{\text{dloss}(\pi)}{\text{d}\pi}$

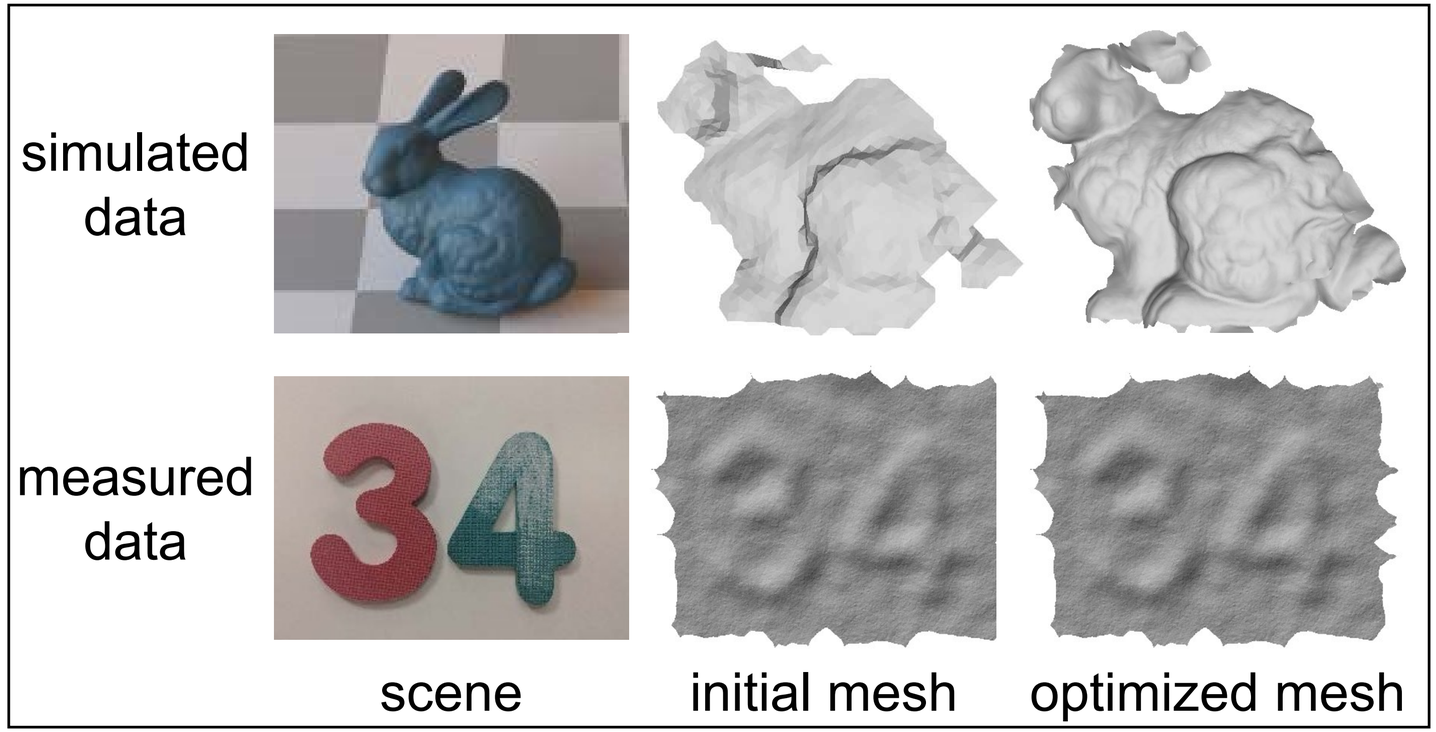
Differentiable rendering

High signal-to-noise ratio is critical

- The extent to which we can improve upon an initialization strongly depends on the signal-to-noise ratio of our measurements

High signal-to-noise ratio is critical

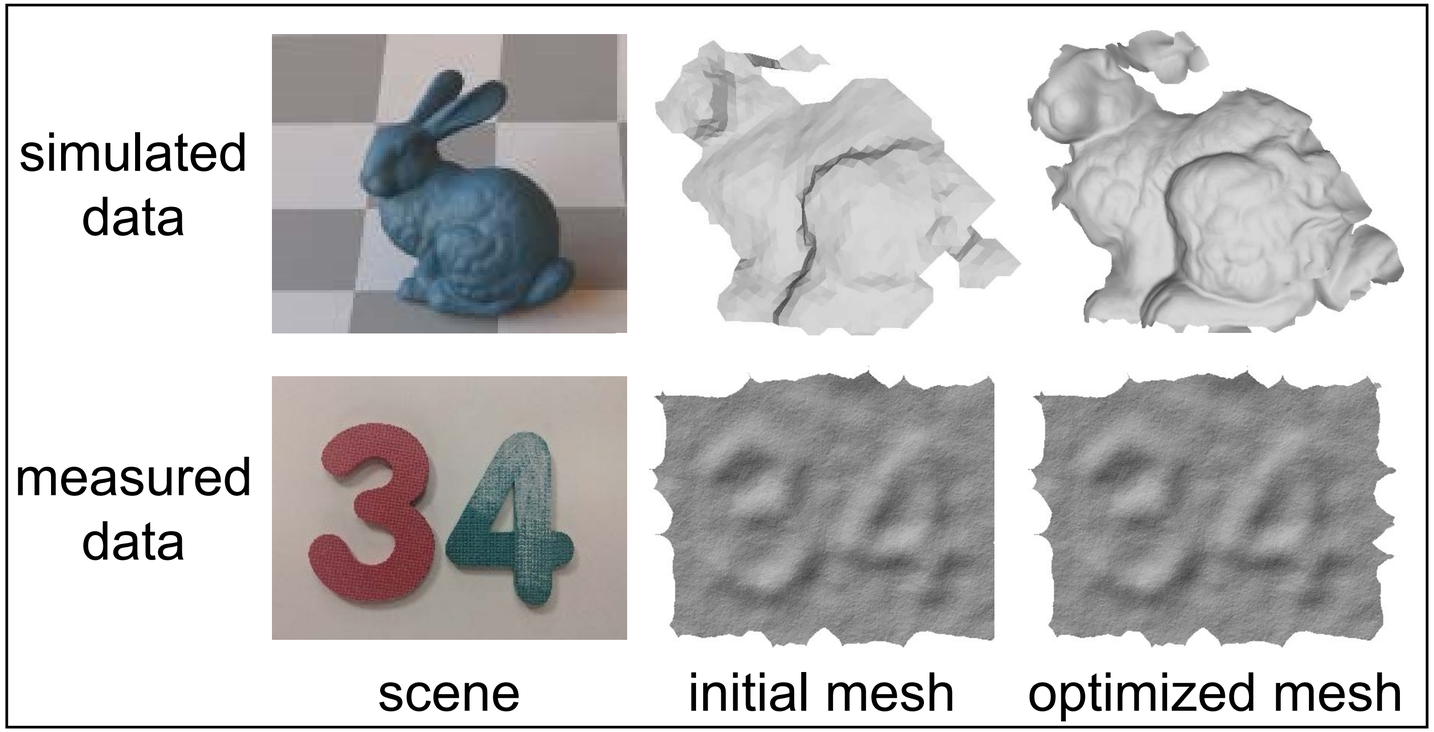
- The extent to which we can improve upon an initialization strongly depends on the signal-to-noise ratio of our measurements



Non-line-of-sight imaging [Tsai et al. 2019]

High signal-to-noise ratio is critical

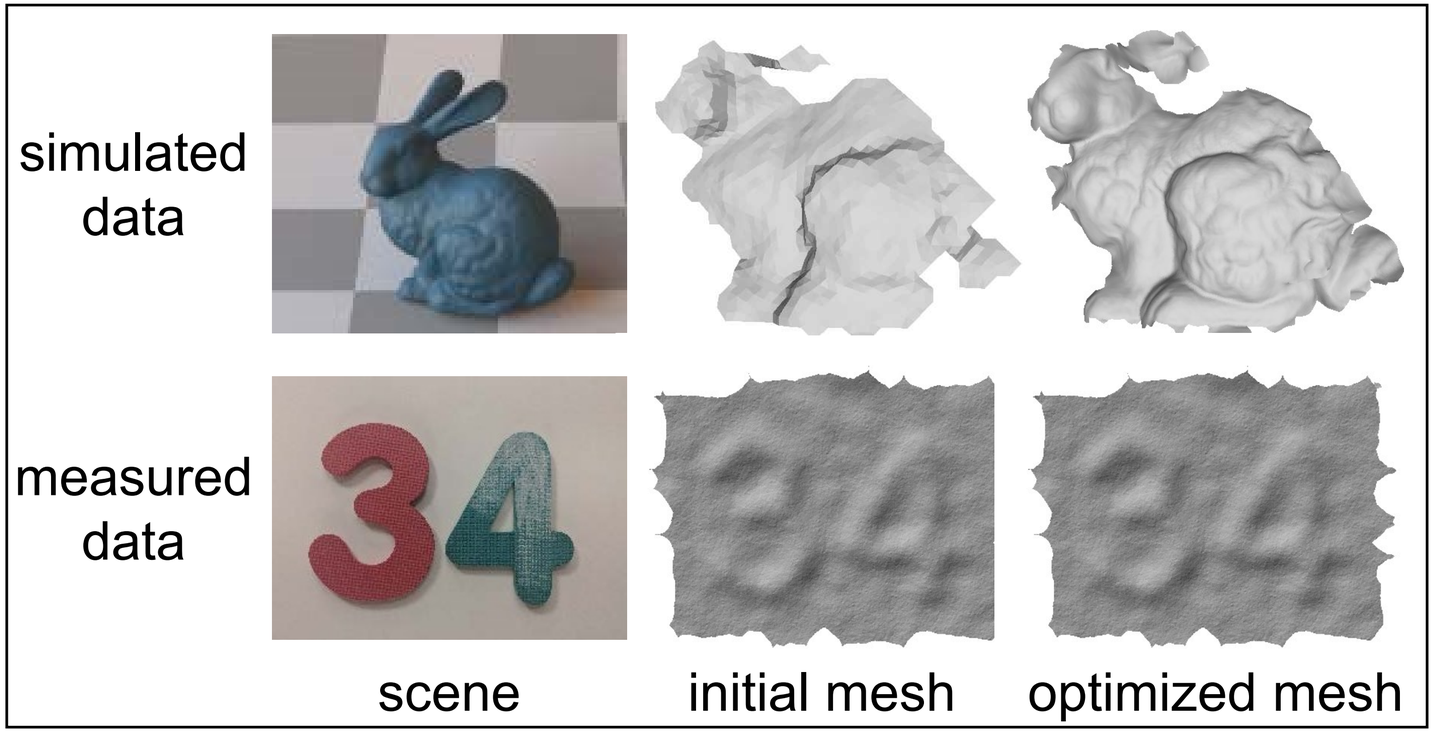
- The extent to which we can improve upon an initialization strongly depends on the signal-to-noise ratio of our measurements
- We need reliable camera models (noise, aberrations, other non-idealities)



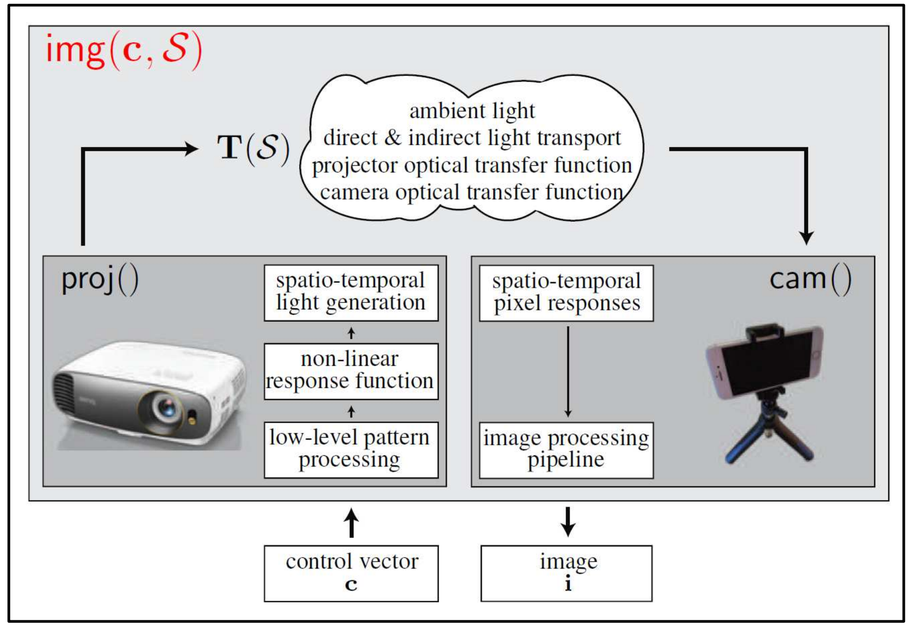
Non-line-of-sight imaging [Tsai et al. 2019]

High signal-to-noise ratio is critical

- The extent to which we can improve upon an initialization strongly depends on the signal-to-noise ratio of our measurements
- We need reliable camera models (noise, aberrations, other non-idealities)



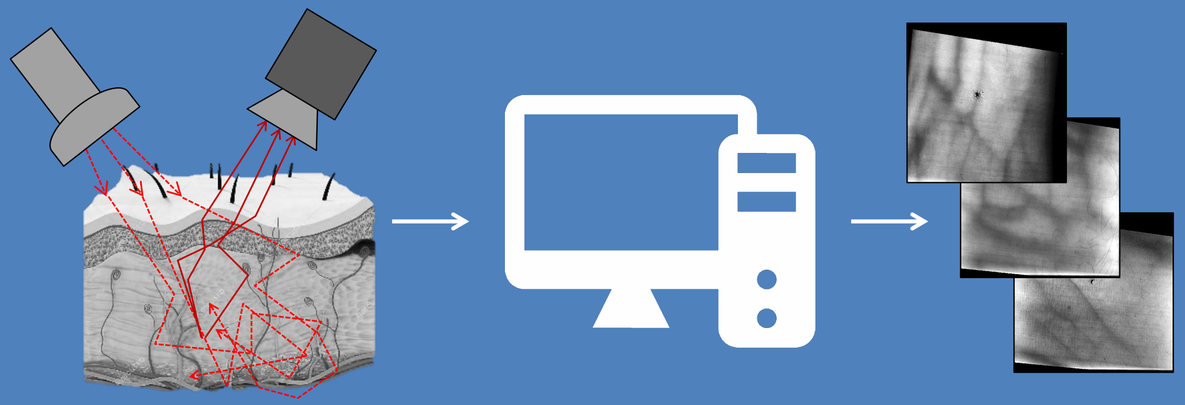
Non-line-of-sight imaging [Tsai et al. 2019]



Optical gradient descent [Chen et al. 2020]

Physics-based rendering and its applications to computational imaging

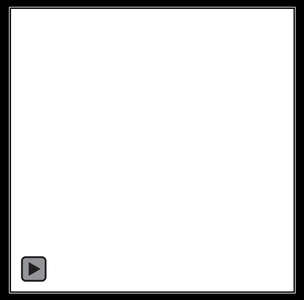
forward rendering



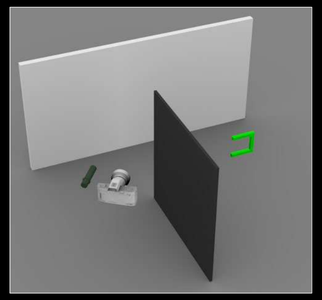
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

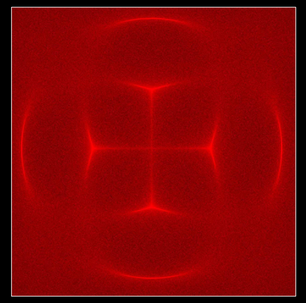
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



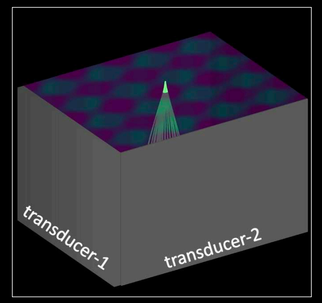
time-of-flight imaging



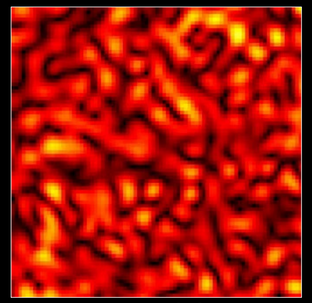
non-line-of-sight imaging



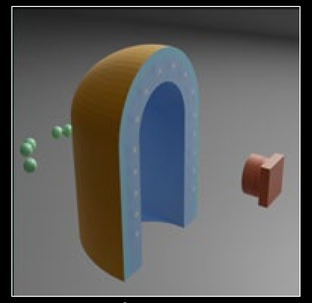
acousto-optic lensing



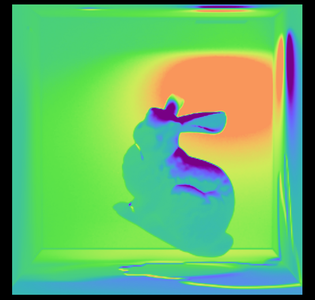
ultrafast light scanning



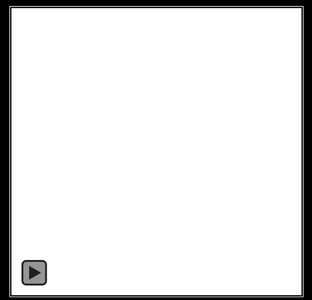
speckle imaging



tactile sensor design



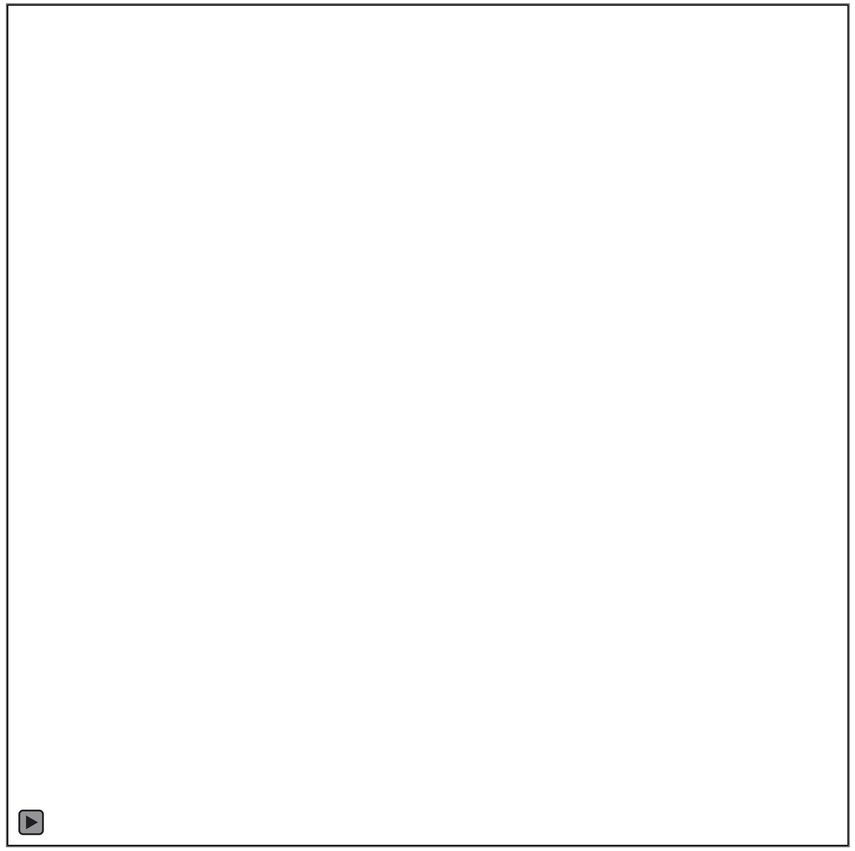
differentiable rendering



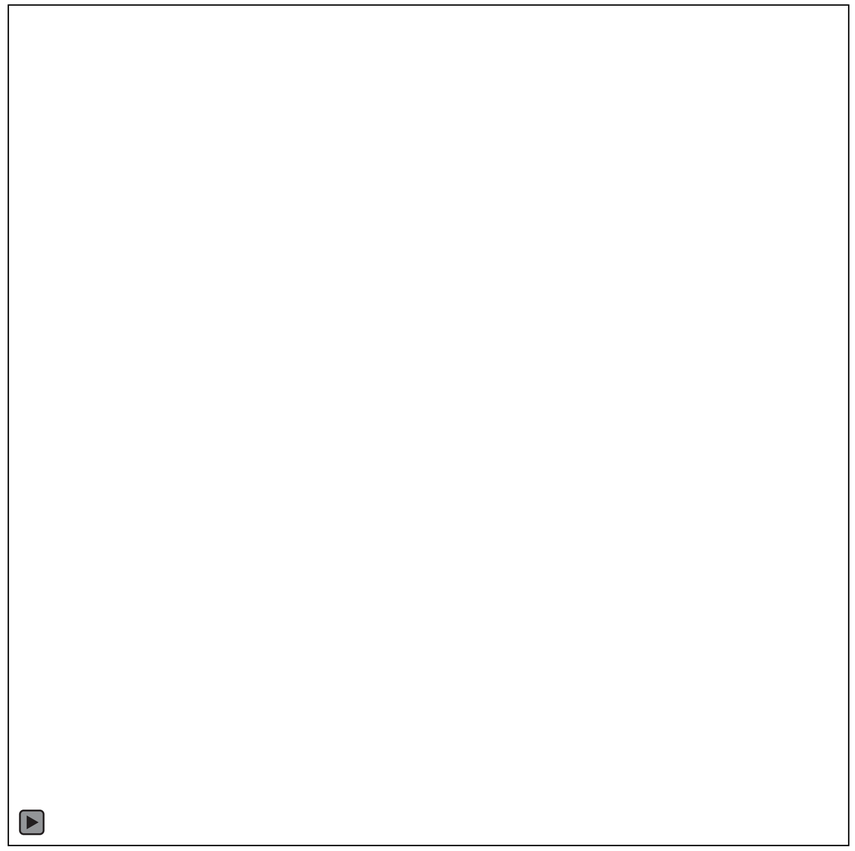
inverse problems

Take-Home Messages

- Great progress has been made in physics-based rendering
  - Capable of handling multiple types of imaging systems beyond RGB cameras (e.g., time-of-flight, sonar, tactile sensors).



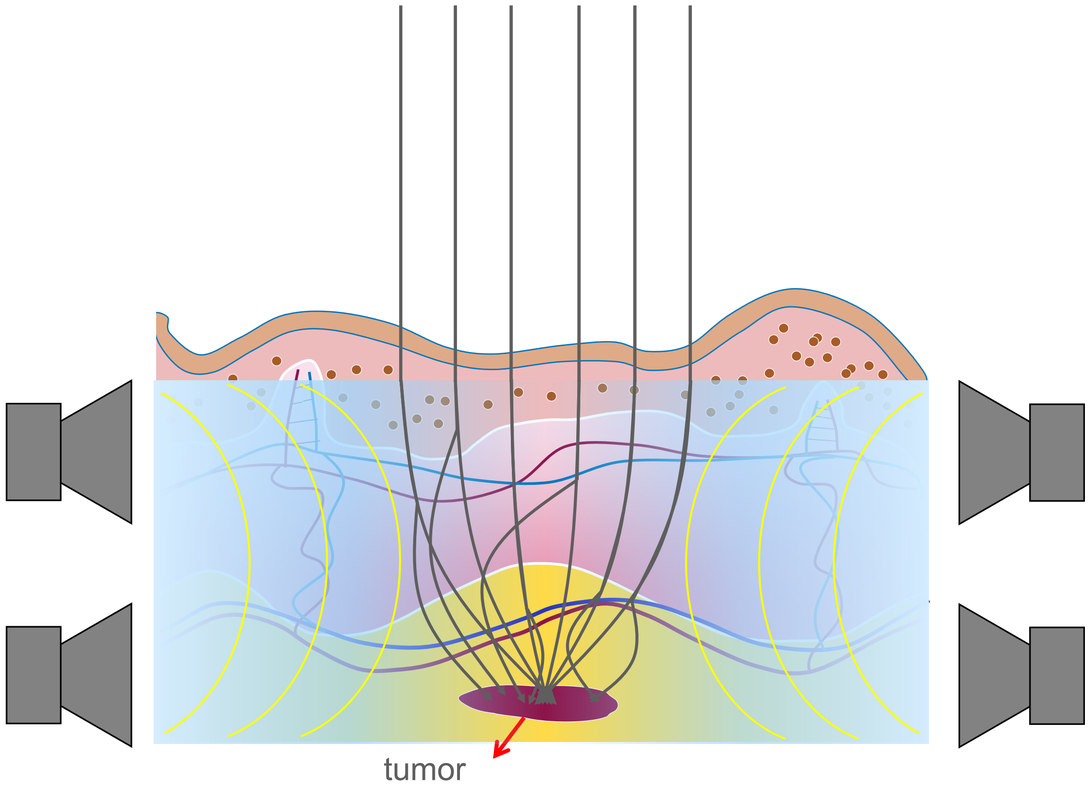
transient rendering of static scene



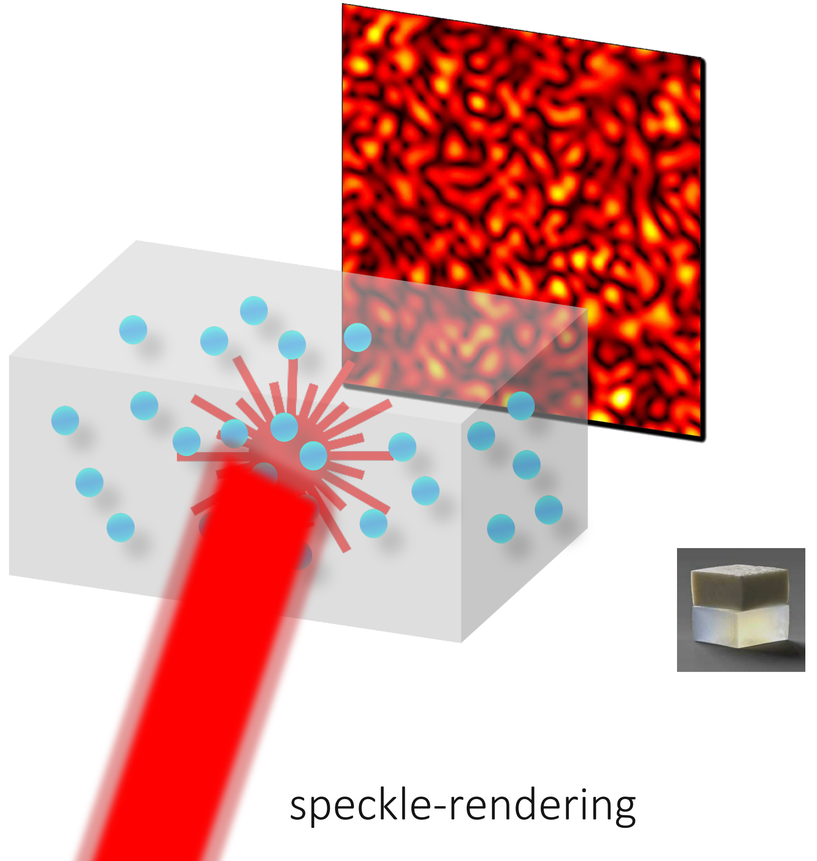
time-gated rendering of dynamic scene

Take-Home Messages

- Great progress has been made in physics-based rendering:
  - Capable of handling multiple types of imaging systems beyond RGB cameras (e.g., time-of-flight, sonar, tactile sensors).
  - Capable of handling more general scene models and light-matter interactions (e.g., speckle, continuous refraction and scattering).



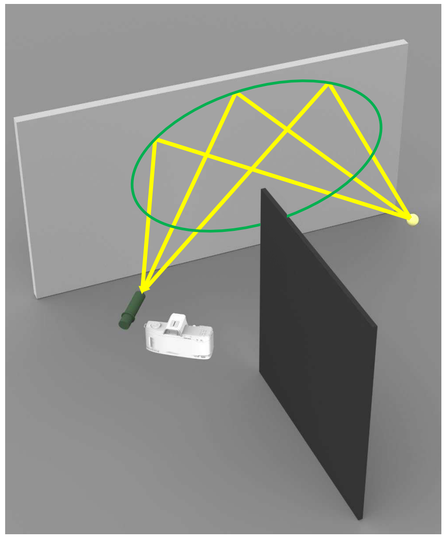
continuous refraction



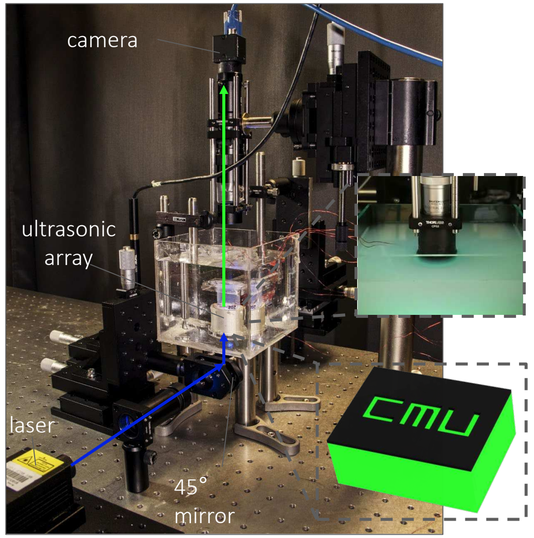
speckle-rendering

Take-Home Messages

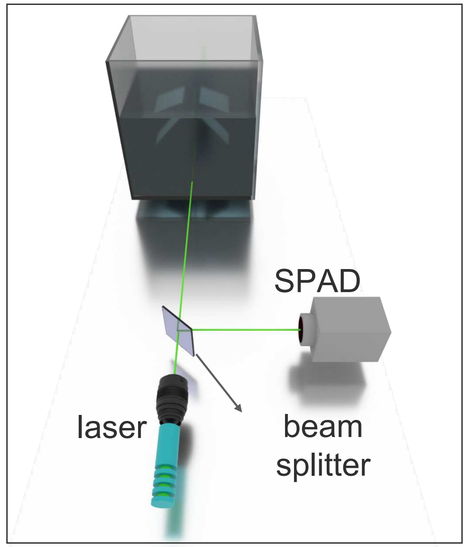
- Great progress has been made in physics-based rendering
  - Capable of handling multiple types of imaging systems beyond RGB cameras (e.g., time-of-flight, sonar, tactile sensors).
  - Capable of handling more general scene models and light-matter interactions (e.g., speckle, continuous refraction and scattering).
  - Capable of acting as digital twins for scientific imaging applications.



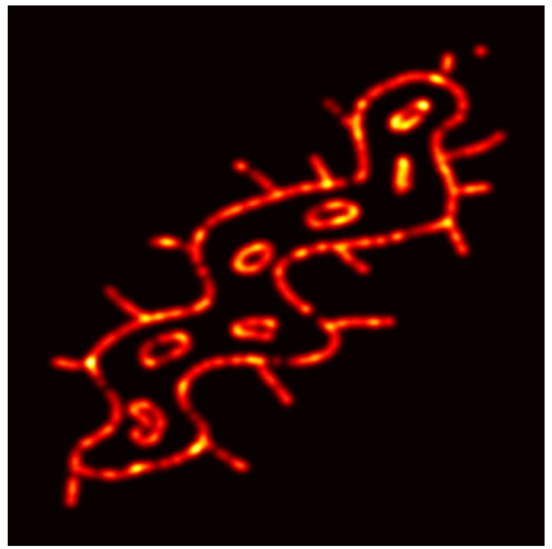
non-line-of-sight imaging



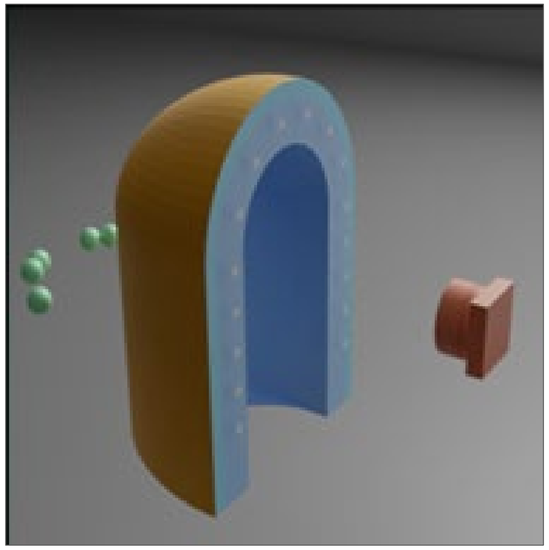
light throughput enhancement



ultrafast light scanners



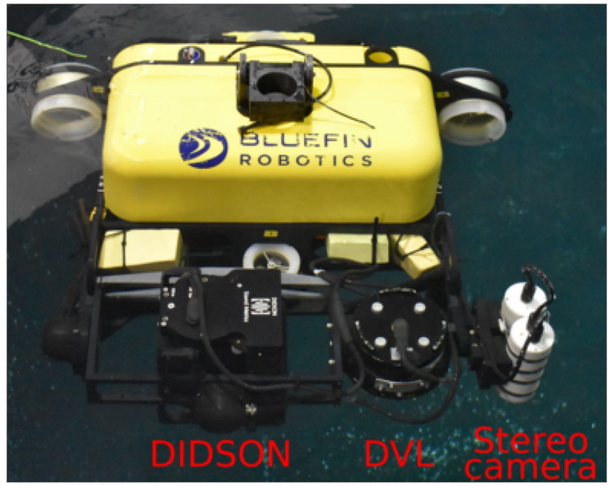
imaging through scattering media



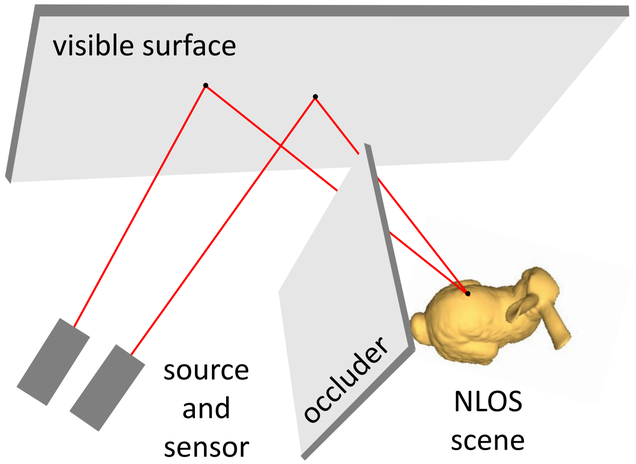
tactile sensor design

Take-Home Messages

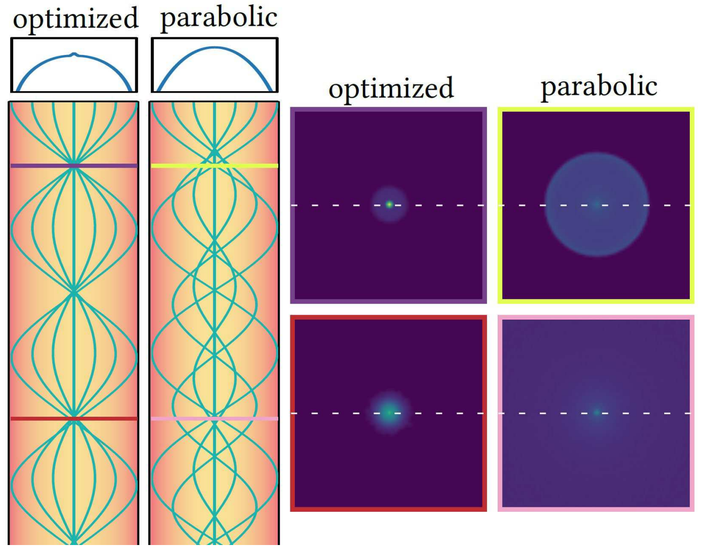
- Great progress has been made in physics-based rendering
  - Capable of handling multiple types of imaging systems beyond RGB cameras (e.g., time-of-flight, sonar, tactile sensors).
  - Capable of handling more general scene models and light-matter interactions (e.g., speckle, continuous refraction and scattering).
  - Capable of acting as digital twins for scientific imaging applications.
  - Capable of differentiation for general inverse rendering problems.



underwater sonar

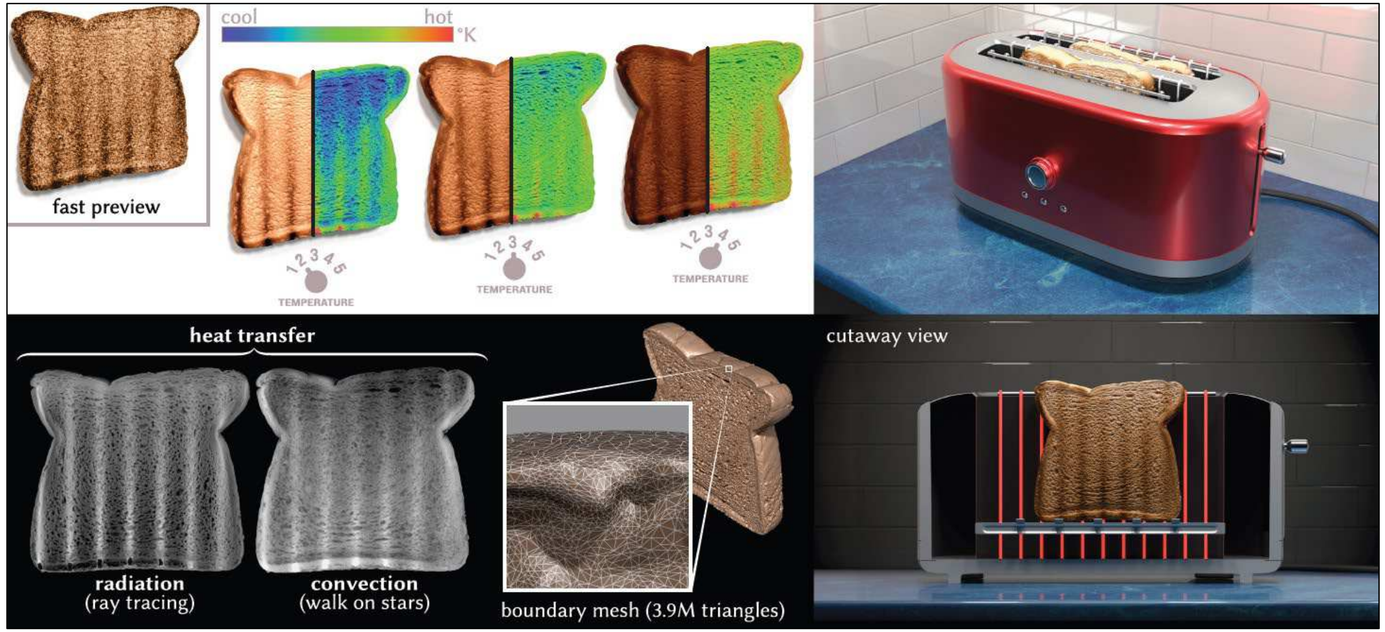


non-line-of-sight imaging



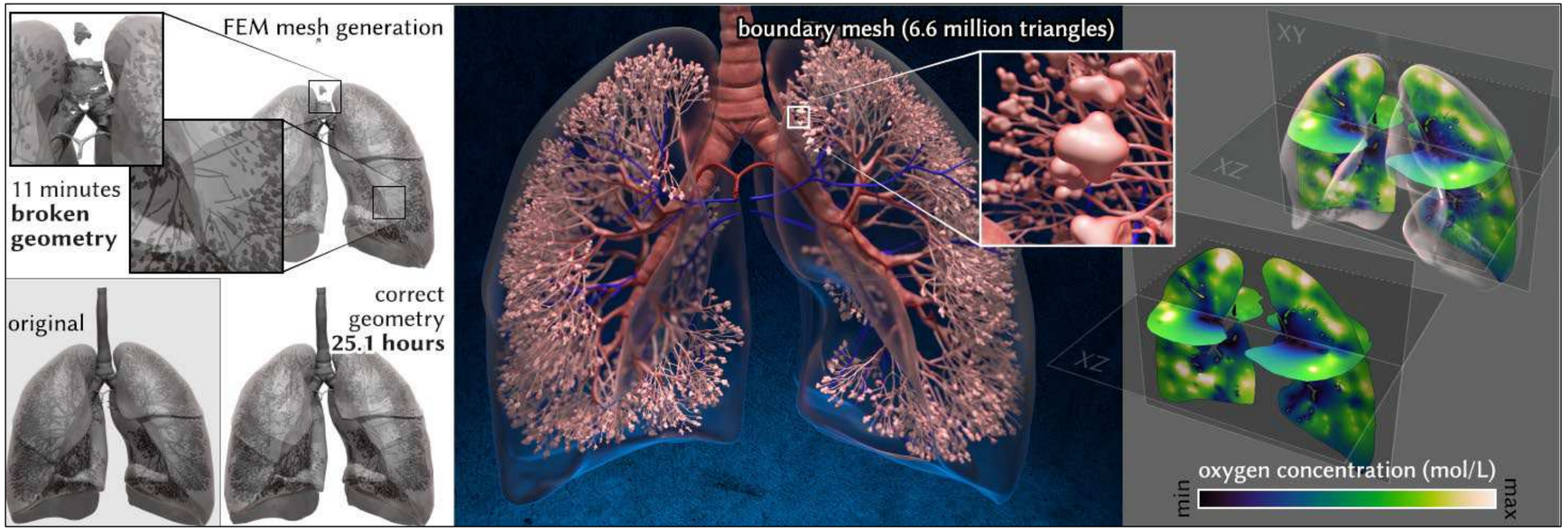
GRIN optic design

Monte Carlo rendering for more general physics and sensing



Simulation of general diffusion processes like heat transfer and oxygen flow

Joint work with
Rohan Sawhney, Bailey Miller,
Keenan Crane
SIGGRAPH 2023




Many thanks to our collaborators

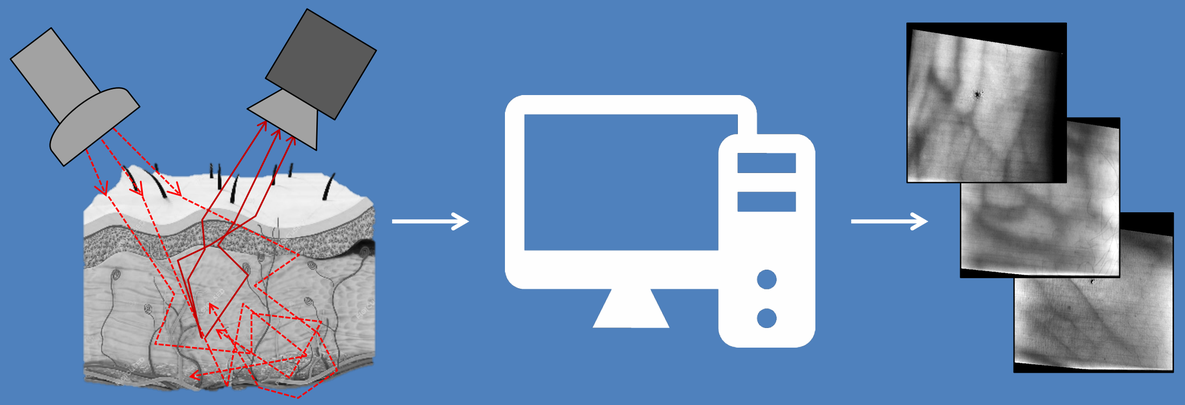
Many thanks to our sponsors

ALFRED P. SLOAN
FOUNDATION

SEE BELOW THE SKIN

Physics-based rendering and its applications to computational imaging

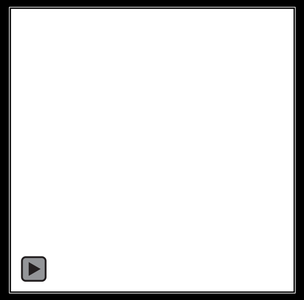
forward rendering



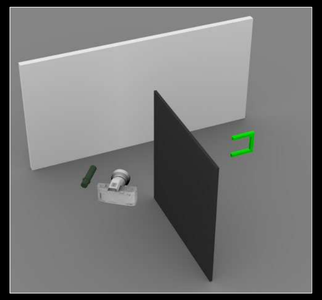
- accurate and efficient simulation
- virtually design sensors, optics, and algorithms

inverse rendering

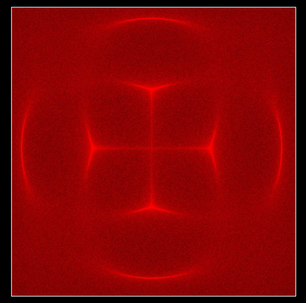
- accurate and efficient differentiable simulation
- tractably solve general inverse problems



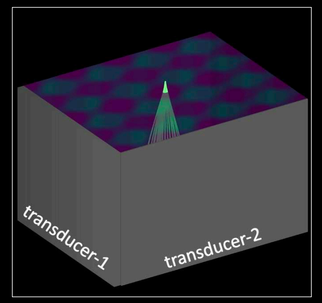
time-of-flight imaging



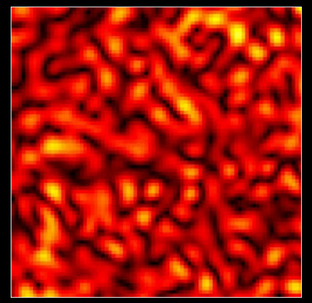
non-line-of-sight imaging



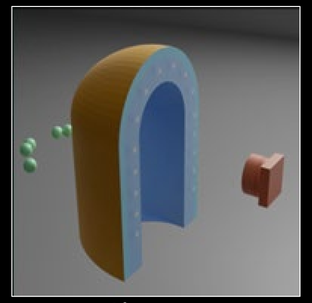
acousto-optic lensing



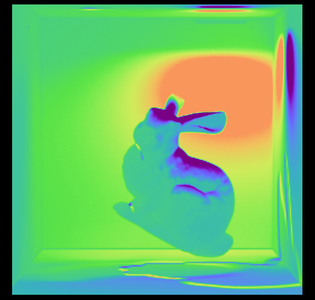
ultrafast light scanning



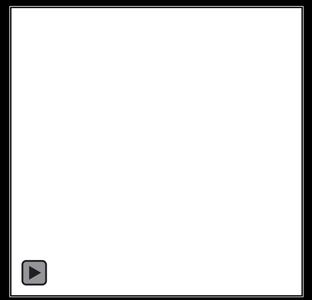
speckle imaging



tactile sensor design



differentiable rendering



inverse problems