Introduction to Lighting and Rendering

Zhi-Hao Lin

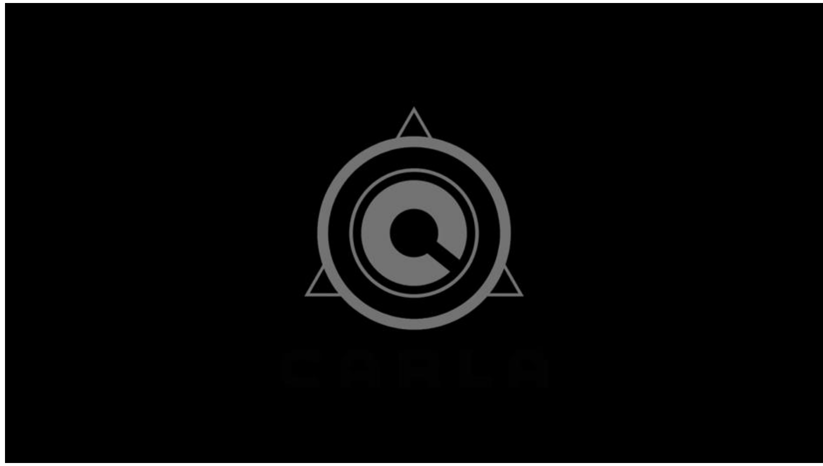
Creating realistic contents is CRUCIAL

MAY CONTAIN CONTENT INAPPROPRIATE FOR CHILDREN

Visit esrb.org for rating information

Video game

AR/VR

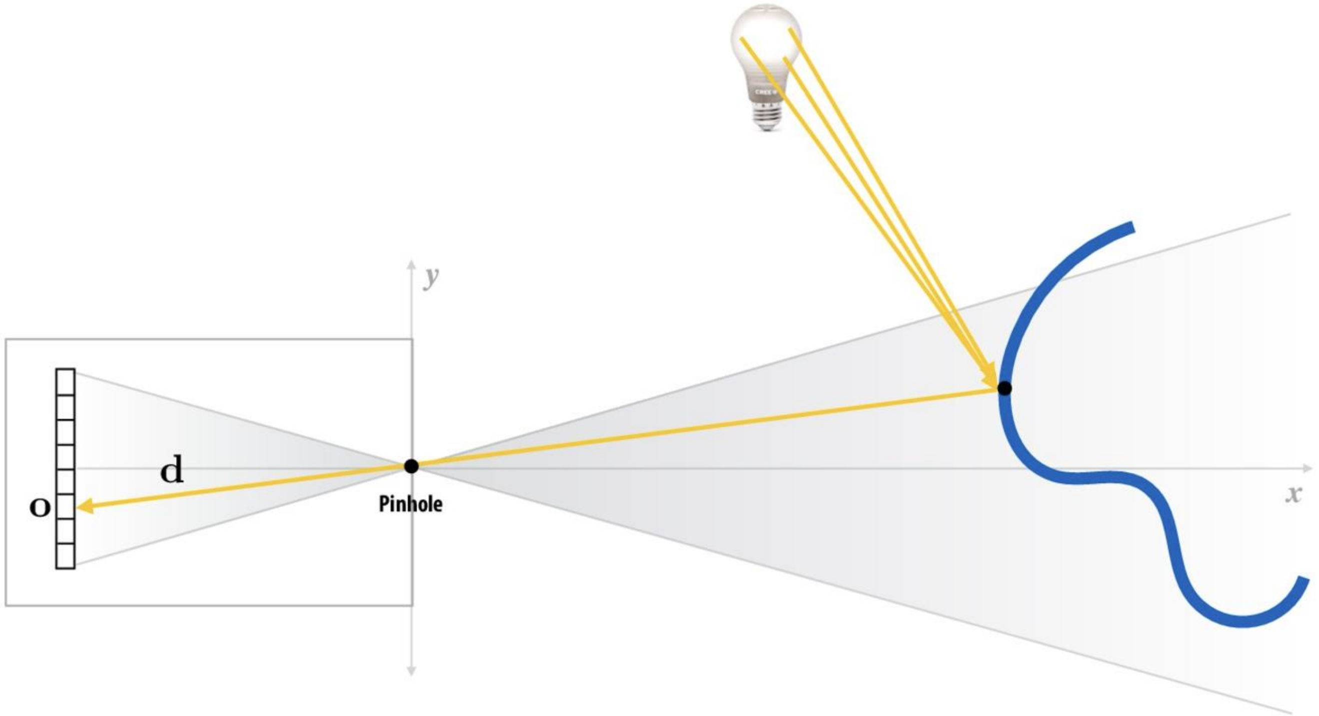


Self-driving simulation

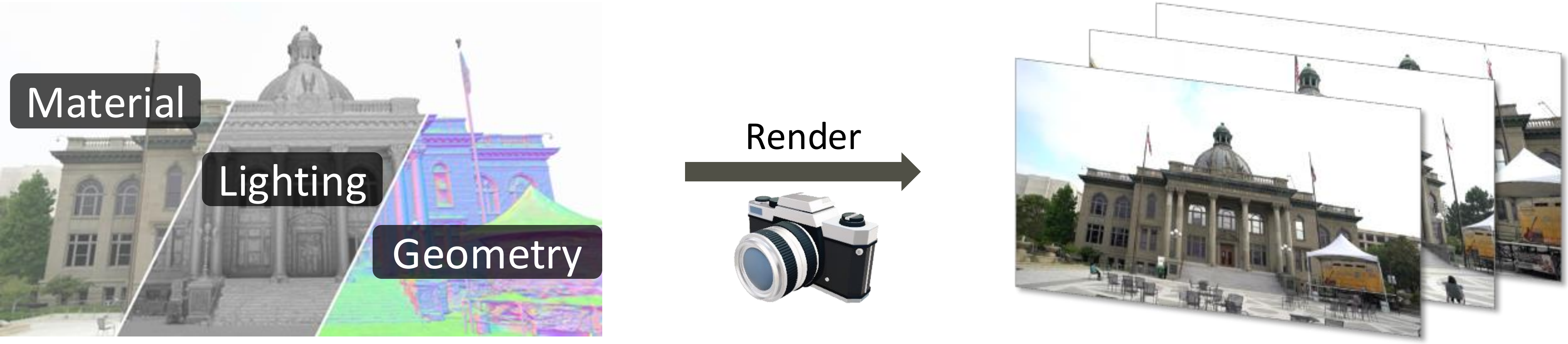
https://www.youtube.com/watch?v=inQelDKULOQ
https://www.youtube.com/watch?v=lY4x85zqoJM
https://www.youtube.com/watch?v=-L9VuPpzVdQ

We are going to talk about …

- What are the basic components of rendering process?
- How to render an image?
- What is inverse rendering? Why is it challenging?



How to render 2D from 3D?



3D scene

2D images/videos

Photo credit:
Neuralangelo: High-Fidelity Neural Surface Reconstruction (CVPR’23)
Tanks and Temples (https://www.tanksandtemples.org/)

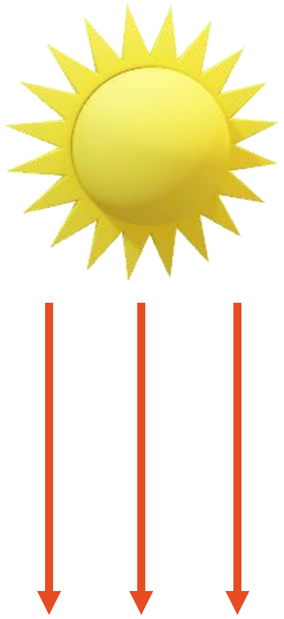
Geometry

In rendering pipeline, the geometry is usually represented as **<u>meshes</u>**

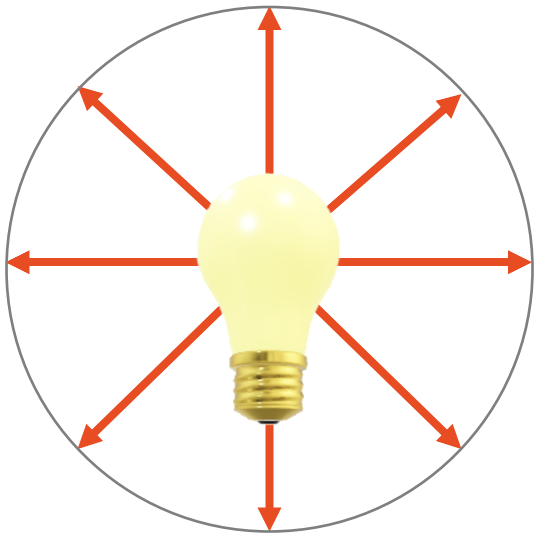
FlexiCubes (ToG, SIGGRAPH 2023)

What papers did we discuss for geometry reconstruction/generation?

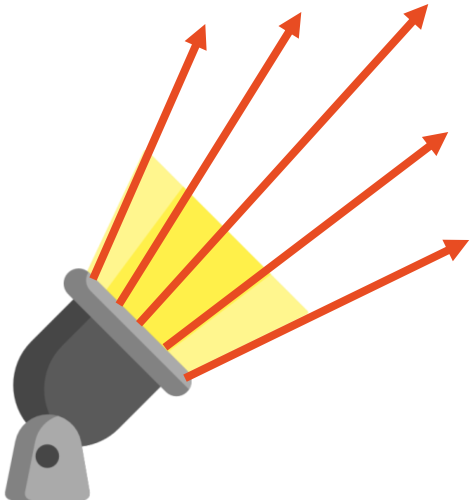
Light sources



Directional light



Point light

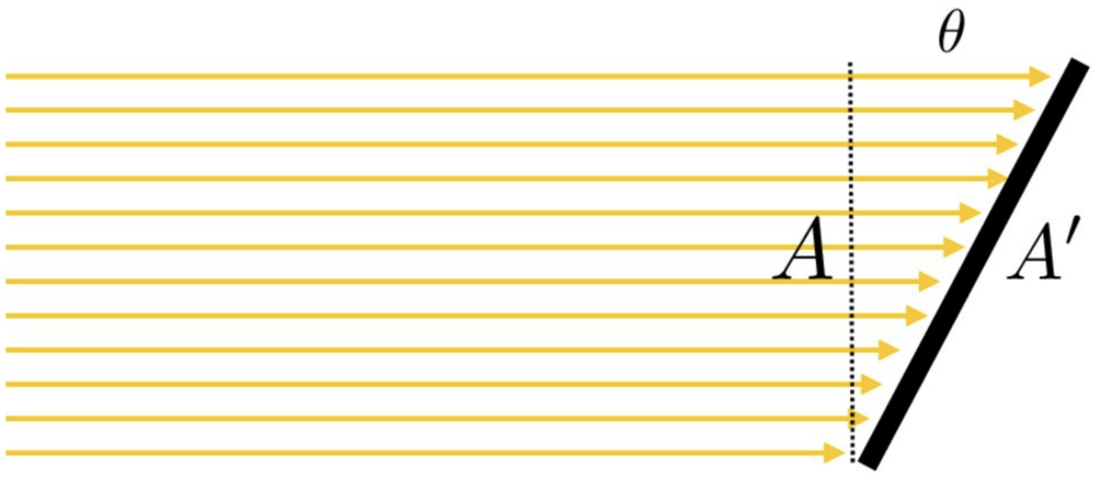


Spotlight

Light direction

Lambert’s Law

Light intensity at surface is proportional to cosine of angle between light direction and surface normal



$$A = A' \cos\theta$$

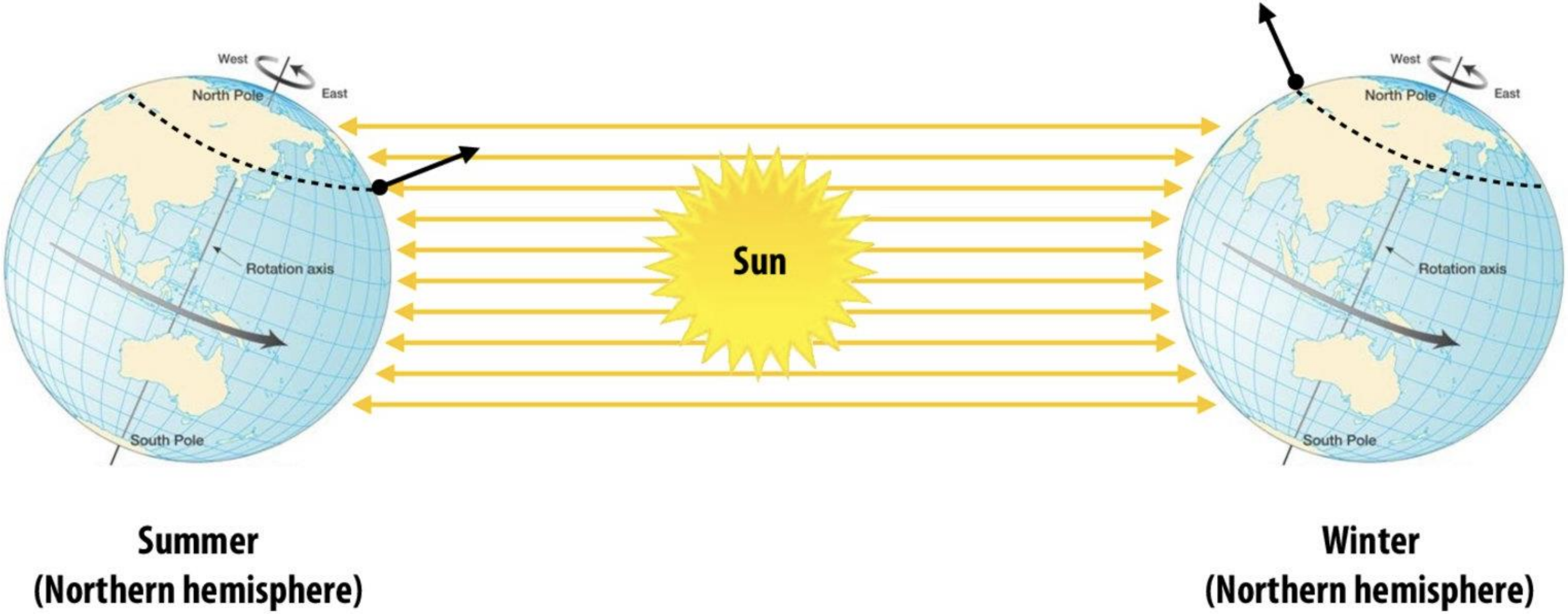
Irradiance

Radiant flux
(energy per time)

$$E = \frac{\Phi}{A'} = \frac{\Phi \cos\theta}{A}$$

Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

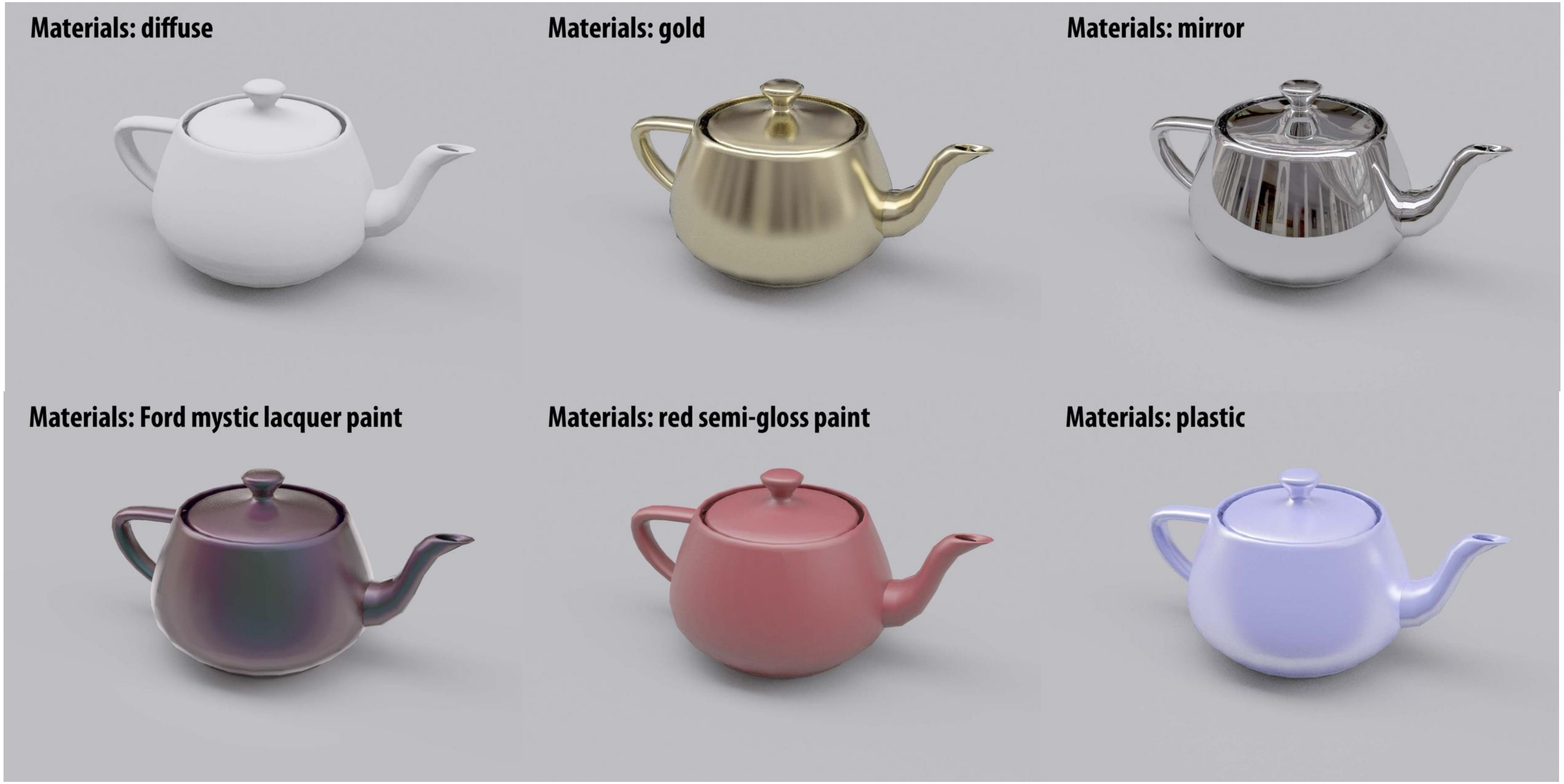
Why do we have seasons?



Earth's axis of rotation: ~23.5° off axis

Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

Material

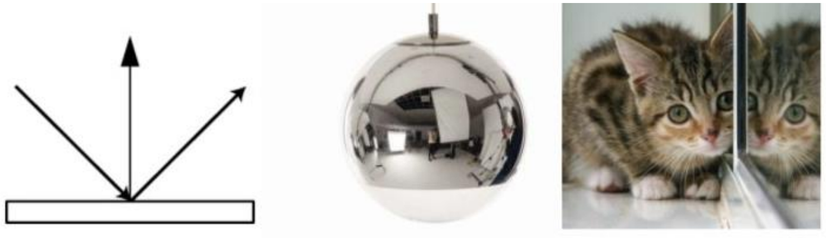


Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

Some basic reflection functions

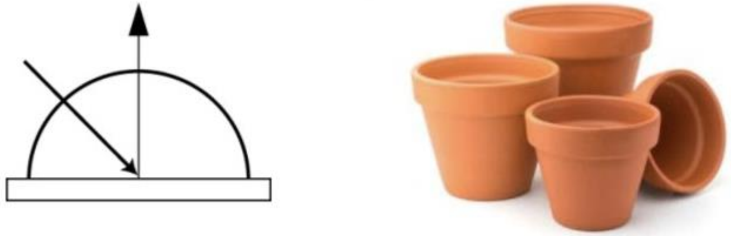
- **Ideal specular**

  Perfect mirror



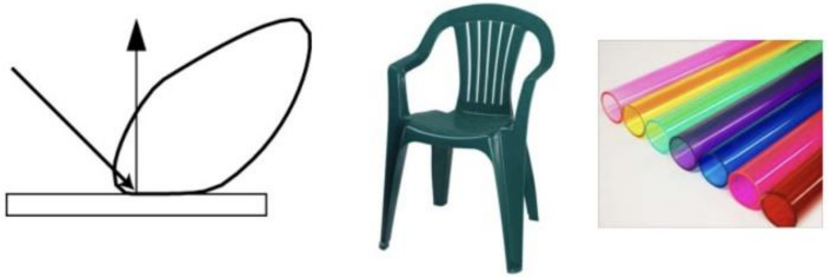
- **Ideal diffuse**

  Uniform reflection in all directions



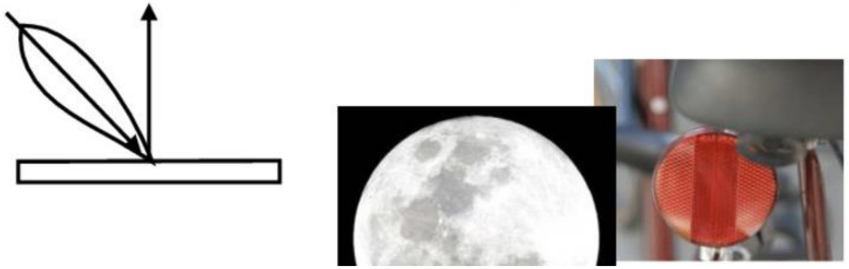
- **Glossy specular**

  Majority of light distributed in reflection direction



- **Retro-reflective**

  Reflects light back toward source



Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

BRDF

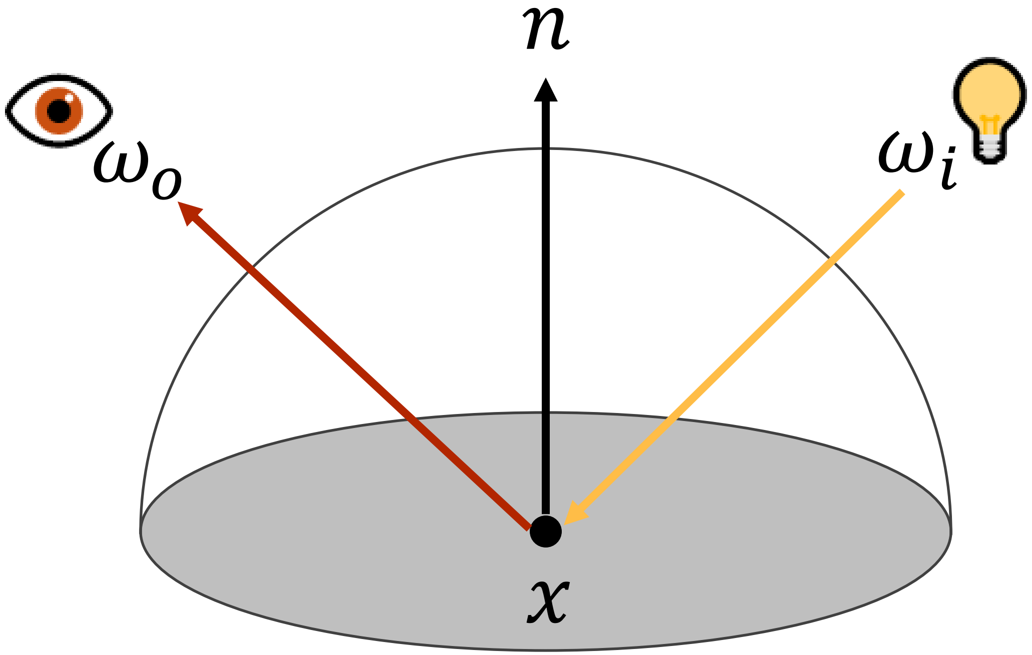
Bidirectional Reflectance Distribution Function

Encodes behavior of light that **<u>bounces off</u>** surface

$$f(\omega_i, \omega_o) = \frac{L_o(\omega_o)}{L_i(\omega_i)}$$

$$= \frac{\text{Outgoing light in direction } \omega_o}{\text{Incoming light in direction } \omega_i}$$

Helmholtz reciprocity: $f(\omega_1, \omega_2) = f(\omega_2, \omega_1)$
$f(\omega_i, \omega_o) \geq 0$



Can BRDI ehaviors?

More light & material behaviors

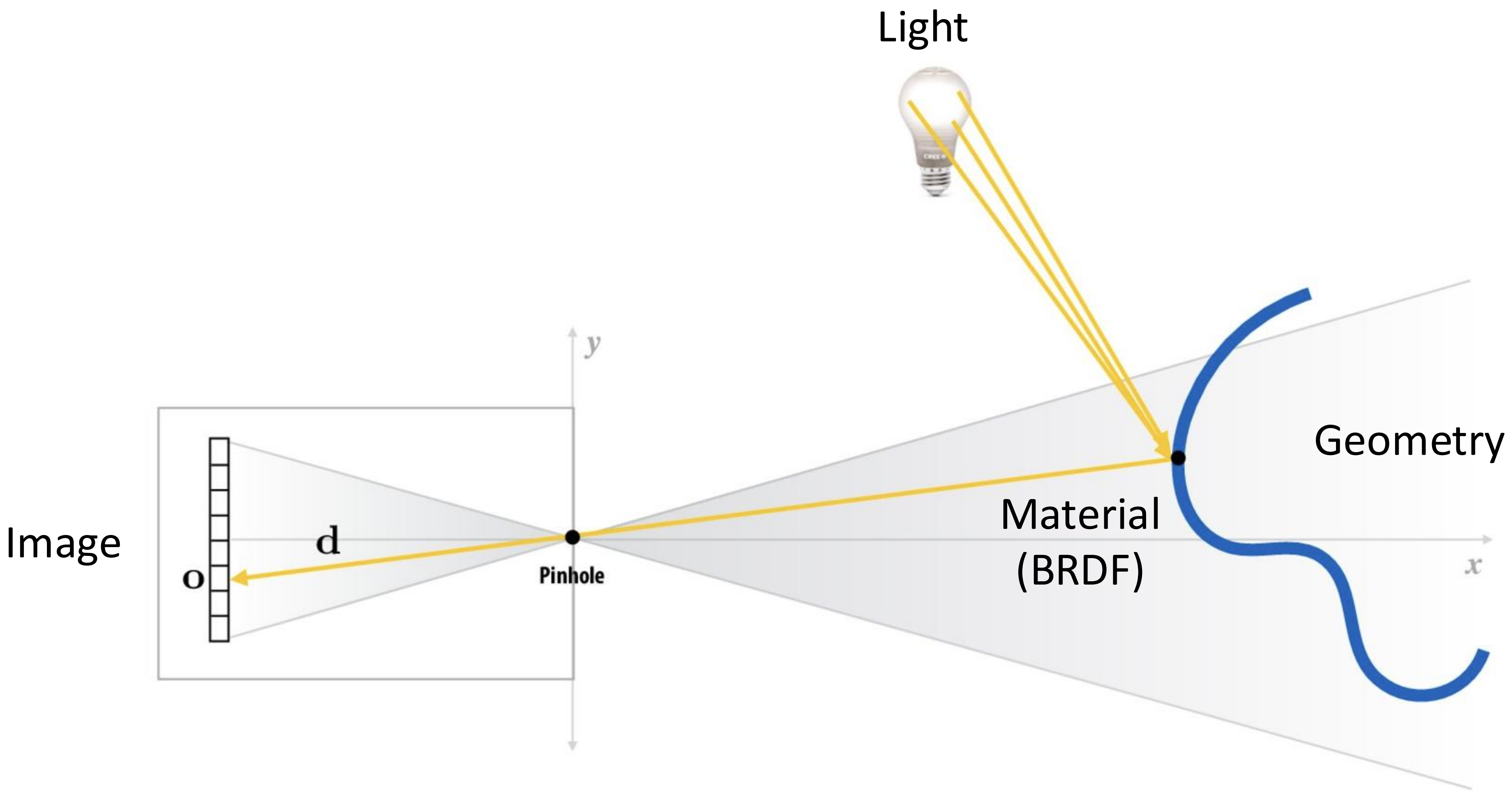
In addition to reflecting off surfaces, light may be transmitted through surfaces

We didn’t cover:

- Refraction
- Fresnel reflection
- Subsurface scattering
- BSDF

[Donner et al 2008]

How to render a pixel?



Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

Rendering equation

$$L_o(x,\omega_o) = L_e(x,\omega_o) +$$

$$\int_\Omega f(x,\omega_i,\omega_o)L_i(x,\omega_i)(\omega_i\cdot n)d\omega_i$$

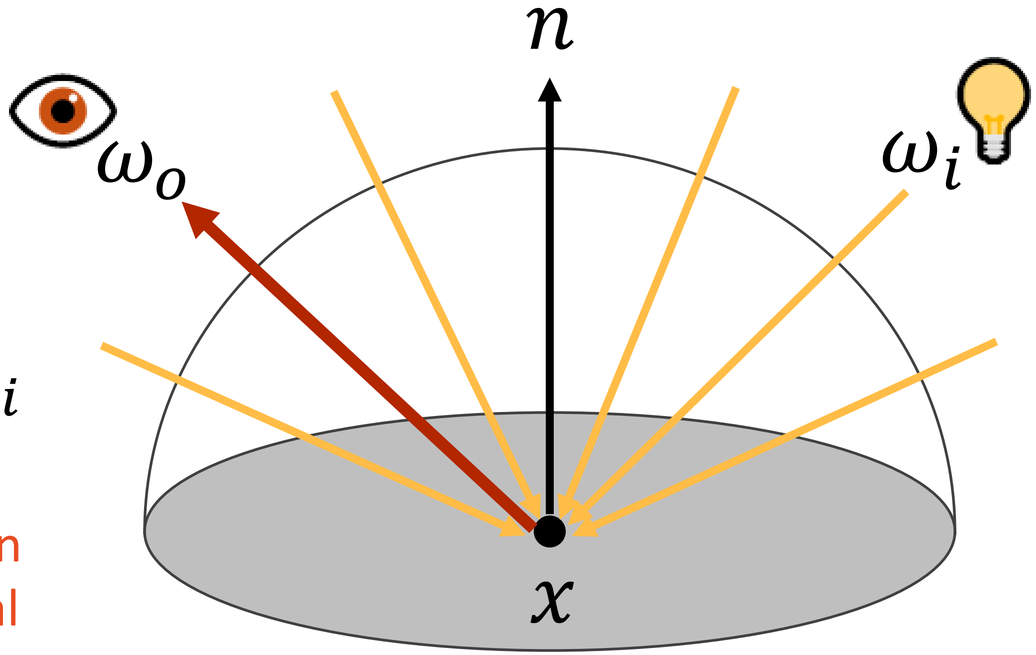
Observed radiance

Emitted radiance (e.g. light source)

Scattering function (e.g. BRDF)

Incoming radiance

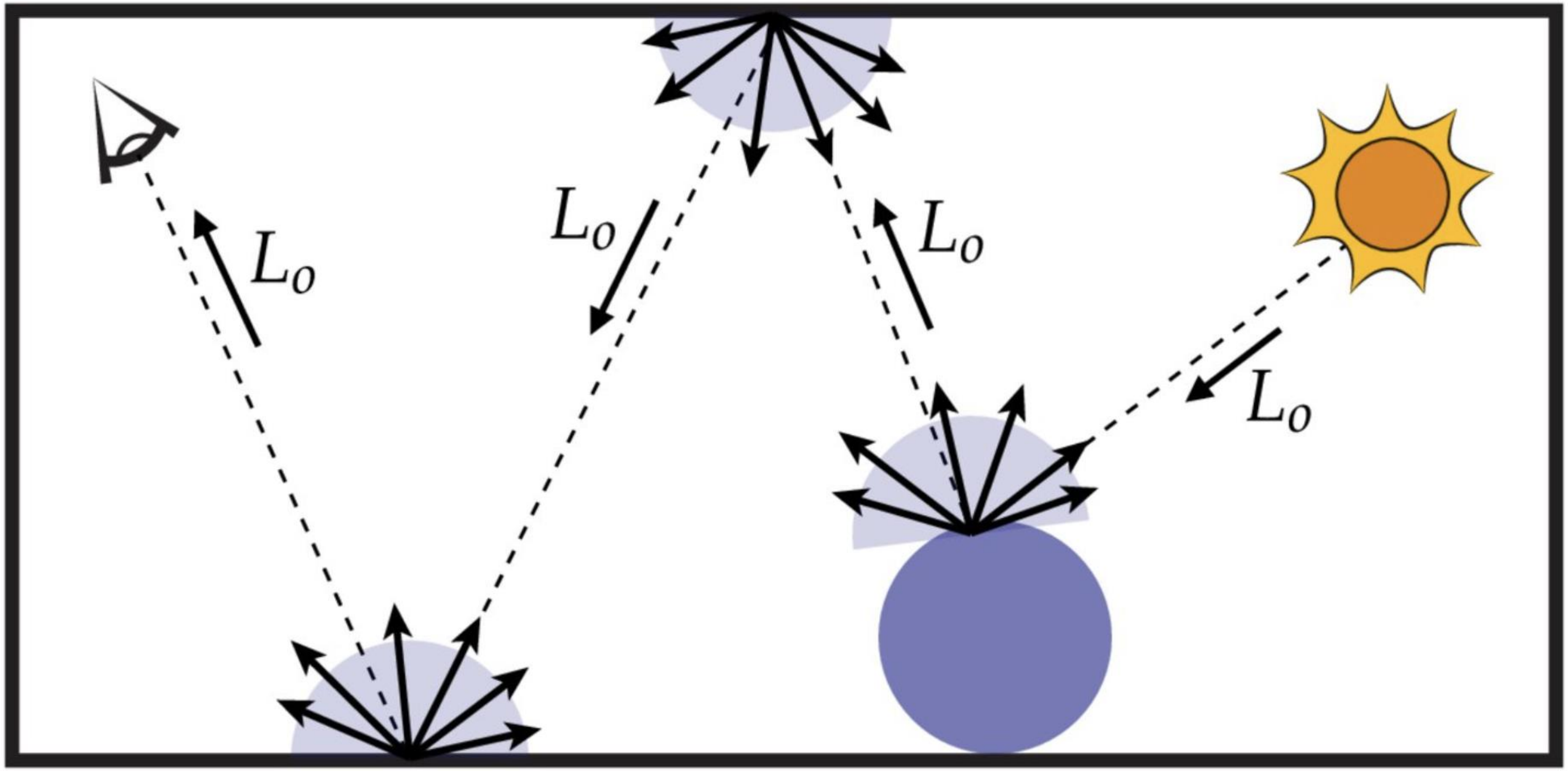
Angle between Light & normal



Rendering equation is **recursive**!

Ray Tracing

- Basic strategy: trace the ray from sensors to light sources!

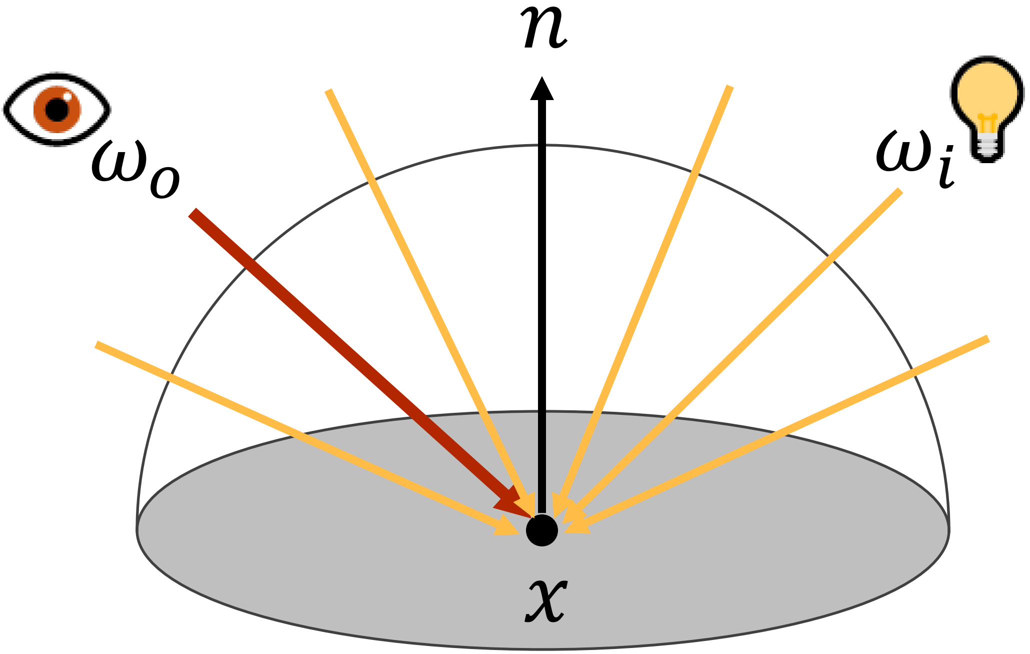


Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

Ray Tracing

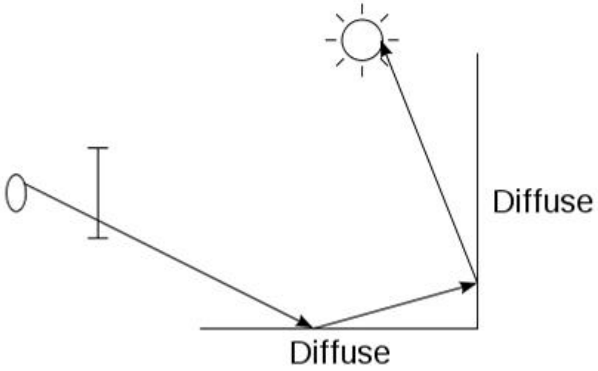
- March the ray until it hits surface
- Sample ray from specific distribution $p(\omega)$(e.g. BRDF)
- Approximate the integral with Monte Carlo integration

$$L_o(x,\omega_o) = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x,\omega_i,\omega_o)L_i(x,\omega_i)(\omega_i\cdot n)}{p(\omega_i)}$$

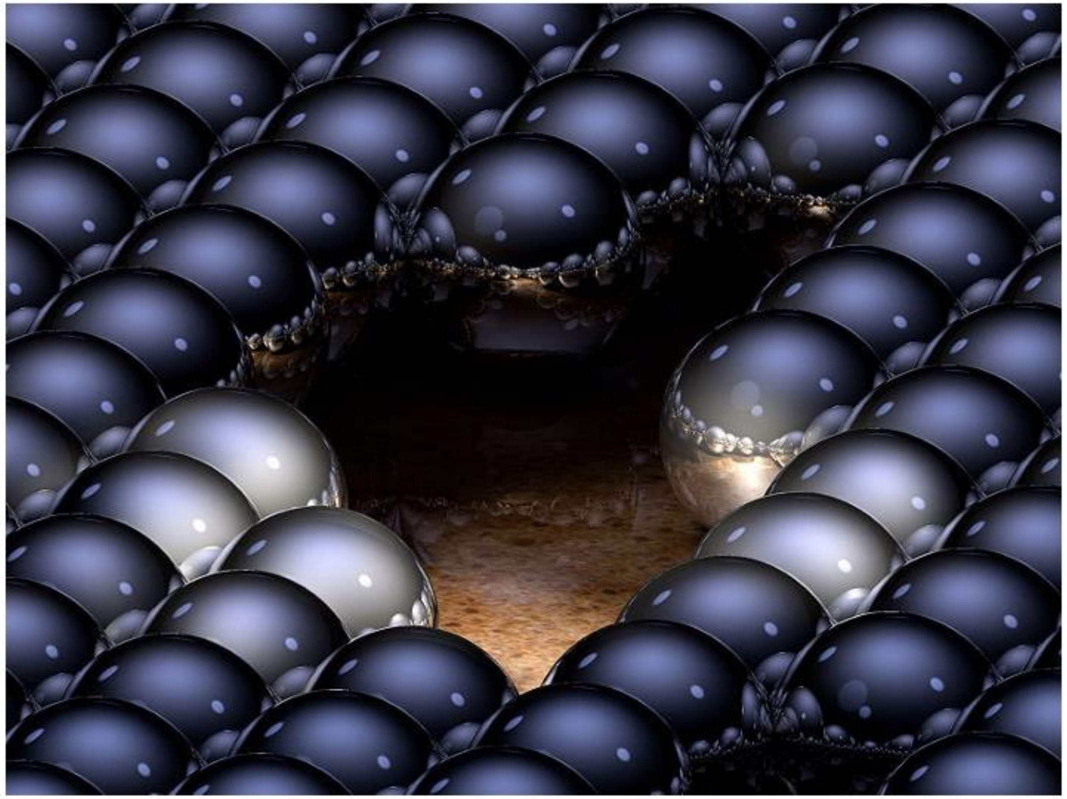


Interreflections

- Reflect light N times before heading to light source



N=2



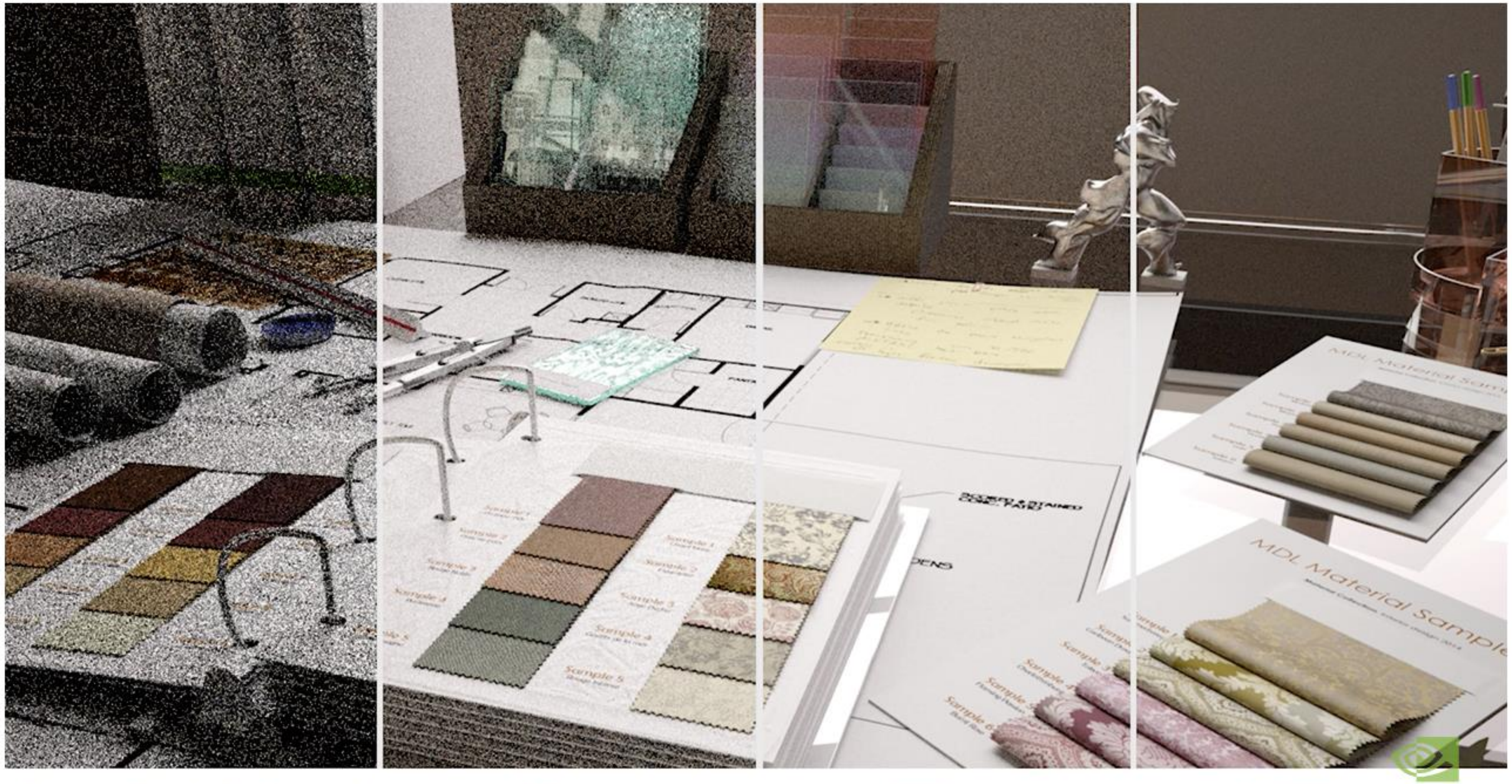
N=16

http://en.wikipedia.org/wiki/Ray_tracing_(graphics)#mediaviewer/File:Ray-traced_steel_balls.jpg

Slide credit: CS445 Computational Photography, Derek Hoiem

Denoising

- Few ray samples lead to noisy images

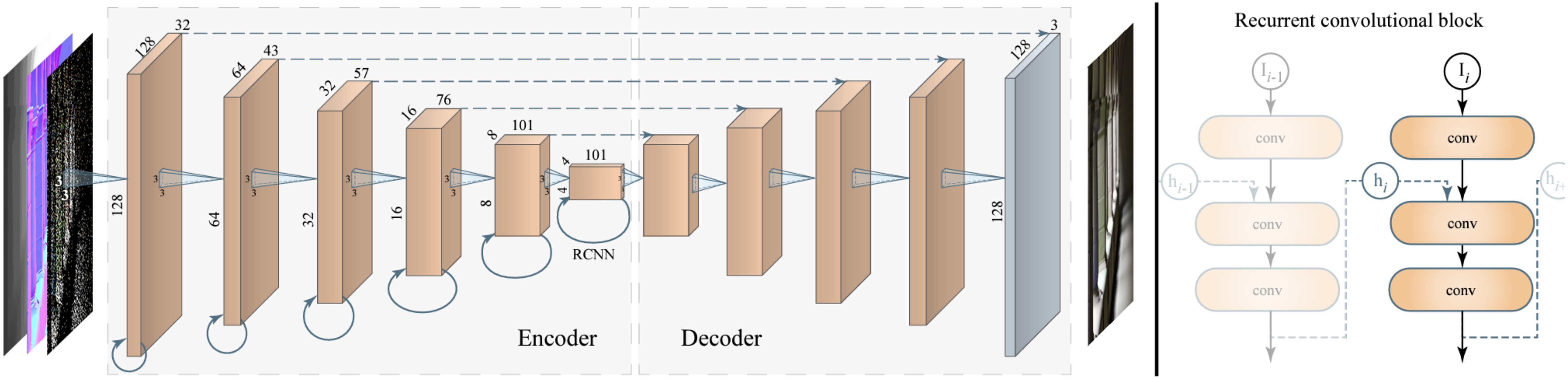


5 samples/pixel (spp) 50 spp 500 spp 5,000 spp

https://developer.nvidia.com/blog/ray-tracing-essentials-part-7-denoising-for-ray-tracing/

Denoising

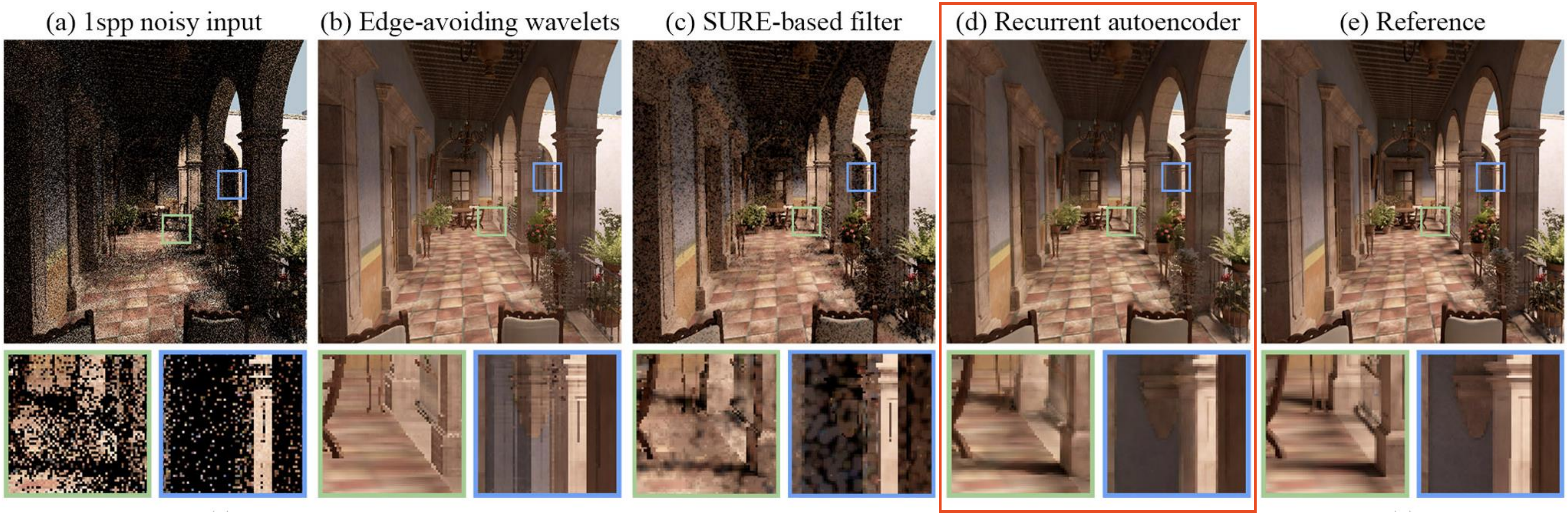
- Based on the information of neighboring pixels, fill in the missing ones
- Deep learning could be used for image denoising



Interactive Reconstruction of Monte Carlo Image Sequences using a Recurrent Denoising Autoencoder (ToG’17)

Denoising

- Based on the information of neighboring pixels, fill in the missing ones
- Deep learning could be used for image denoising



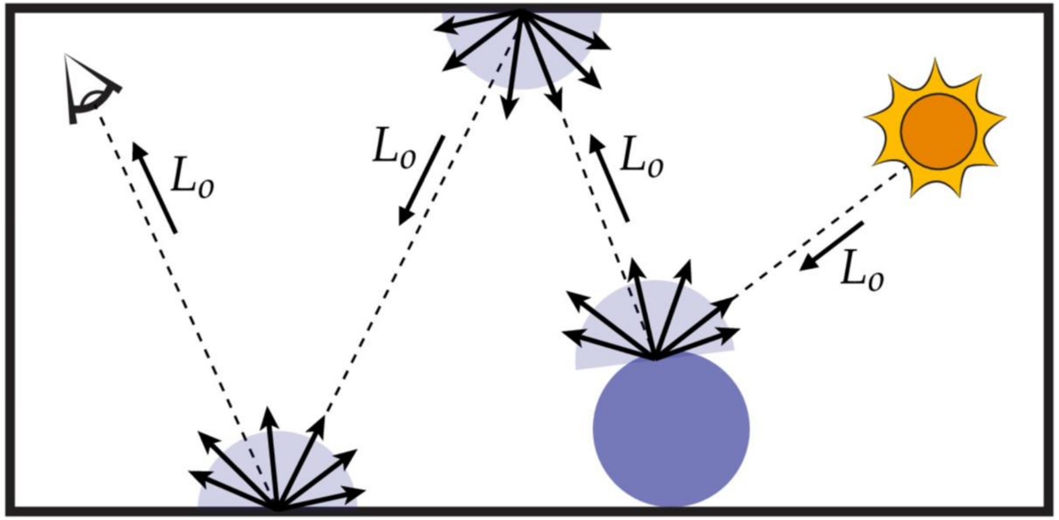
Interactive Reconstruction of Monte Carlo Image Sequences using a Recurrent Denoising Autoencoder (ToG'17)

Ray tracing

- Conceptually simple but hard to do fast

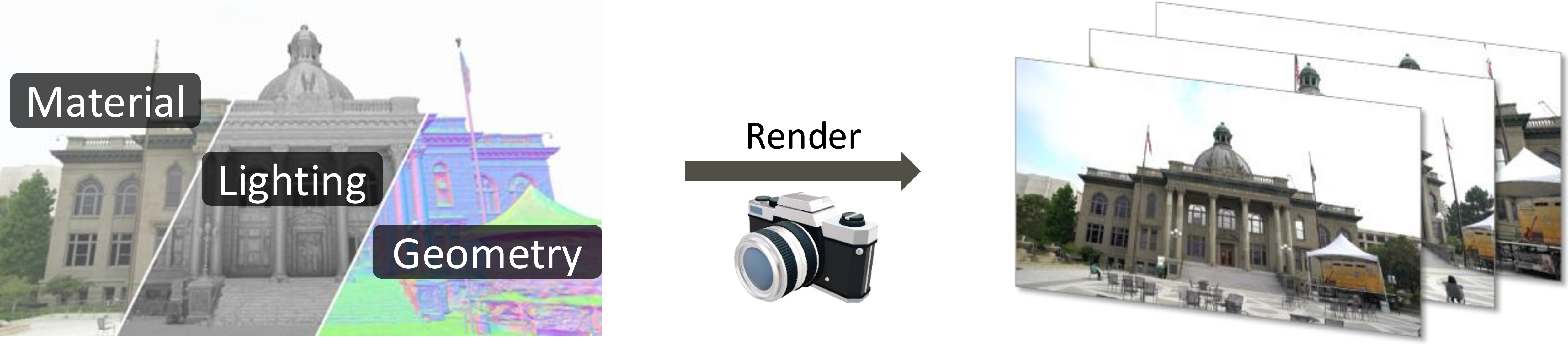
Design choices:

- Ray paths: light → camera vs. camera → light
- How many samples per pixel?
- How to sample the rays?
- When should the rays stop?
- How to denoise the image?
- …



Slide credit: CMU 15-462/662 Computer Graphics, Keenan Crane

How to render 2D from 3D?



3D scene

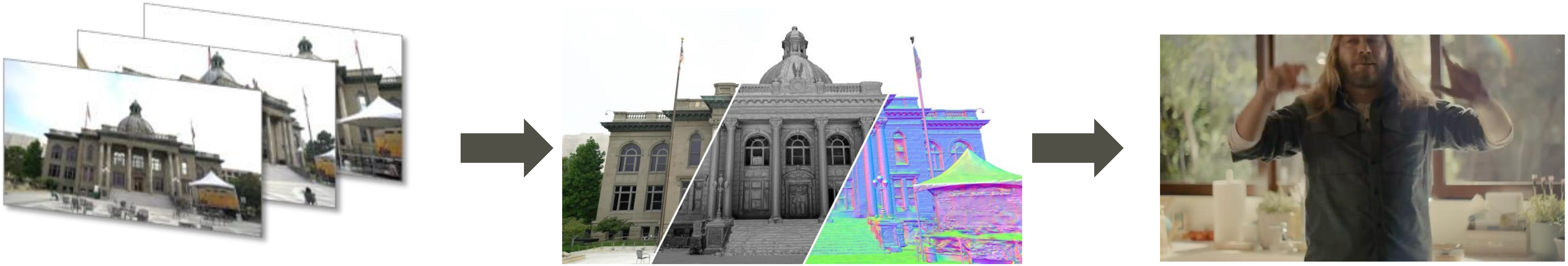
2D images/videos

Photo credit:
Neuralangelo: High-Fidelity Neural Surface Reconstruction (CVPR’23)
Tanks and Temples (https://www.tanksandtemples.org/)

Creating virtual contents is EXPENSIVE

https://www.youtube.com/watch?v=zaqmn55w4IA

Can we reduce the cost?



2D images/videos

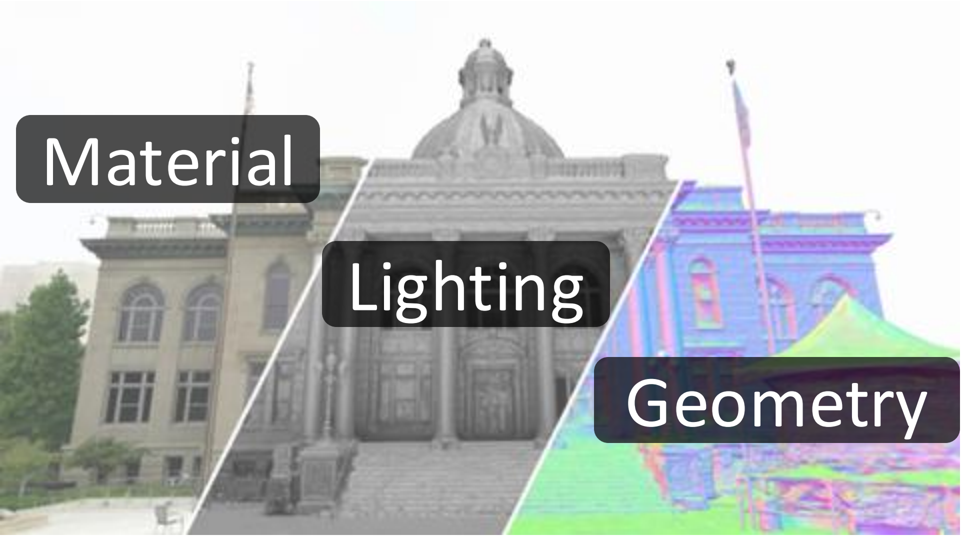
3D scene

Applications (AR/VR)

Photo credit:
Neuralangelo: High-Fidelity Neural Surface Reconstruction (CVPR’23)
Tanks and Temples (https://www.tanksandtemples.org/)
https://www.youtube.com/watch?v=IY4x85zqoJM

Inverse rendering

How to reconstruct 3D from 2D?



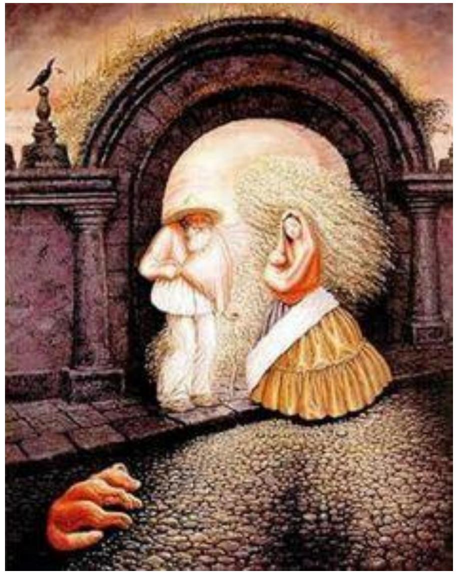
3D scene

Reconstruct

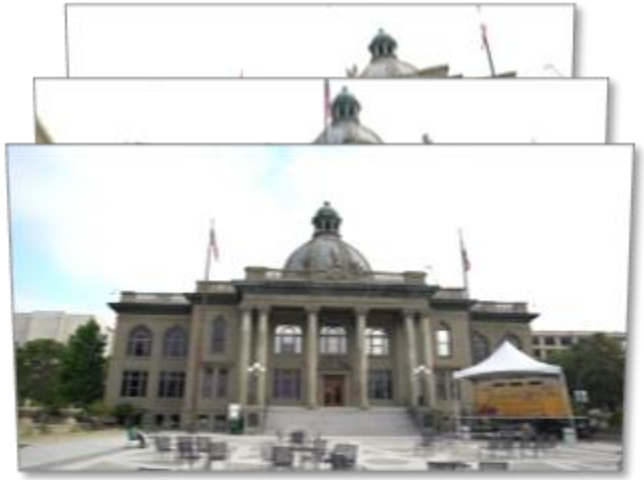
2D images/videos

Photo credit:
Neuralangelo: High-Fidelity Neural Surface Reconstruction (CVPR'23)
Tanks and Temples (https://www.tanksandtemples.org/)

Inverse rendering is CHALLENGING



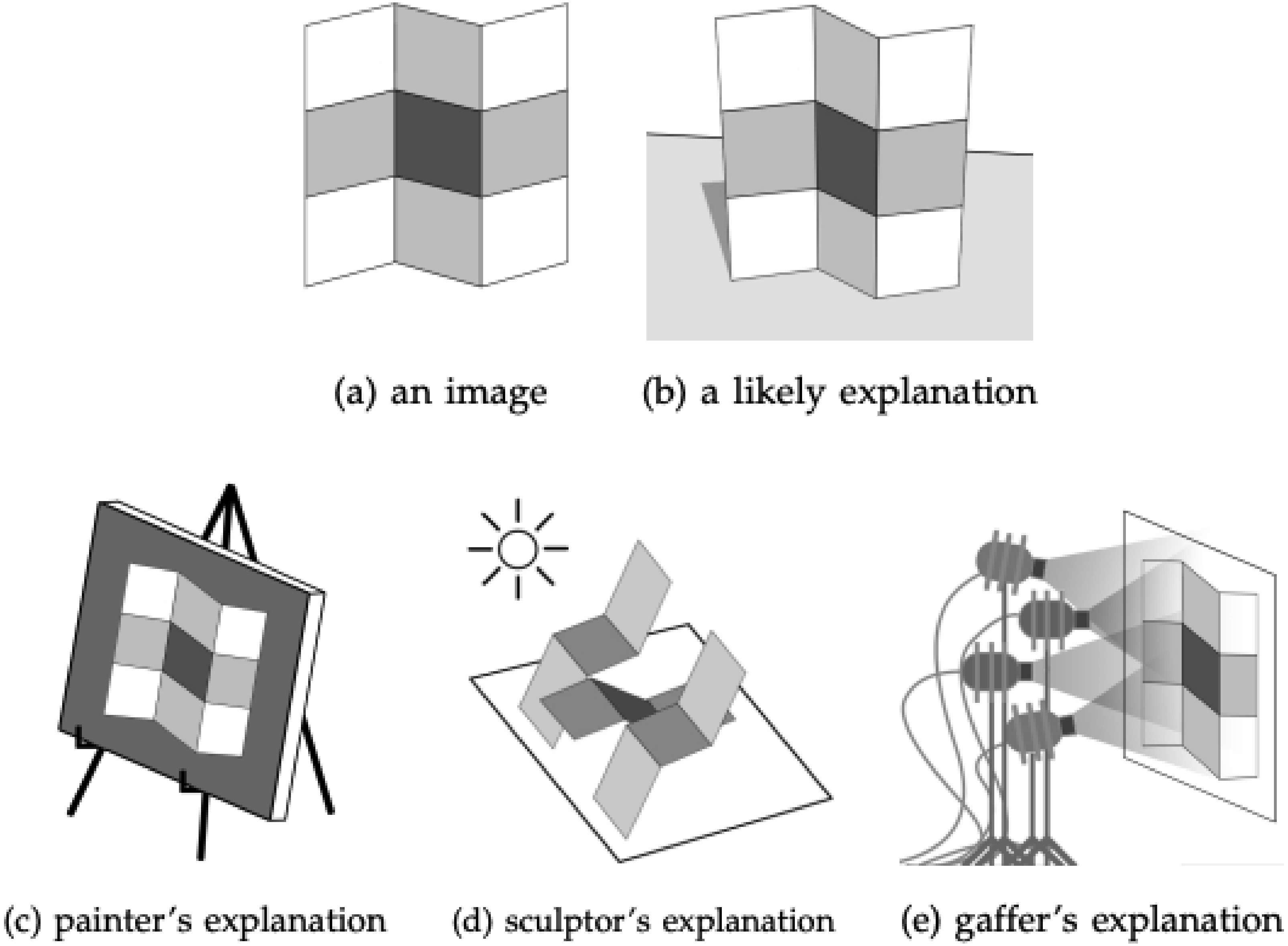
Ill-posed problem



Incomplete observation

Limited real 3D data
Noise in real 2D data

https://bigthink.com/neuropsych/optical-illusions-art-science/
Tanks and Temples (https://www.tanksandtemples.org/)



(a) an image

(b) a likely explanation

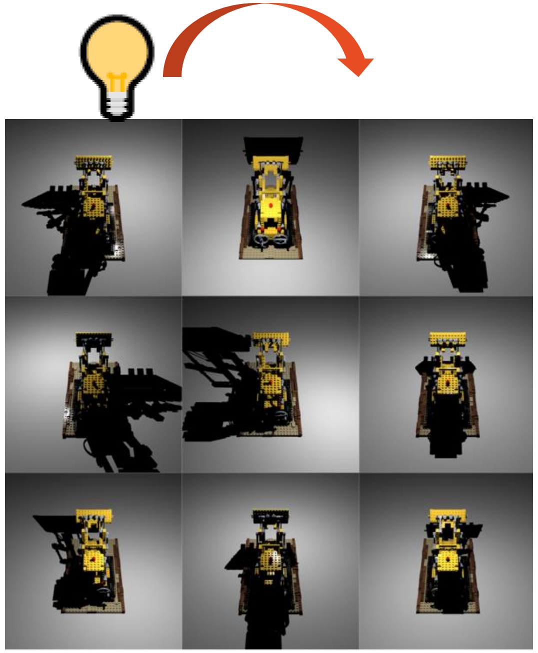
(c) painter's explanation

(d) sculptor's explanation

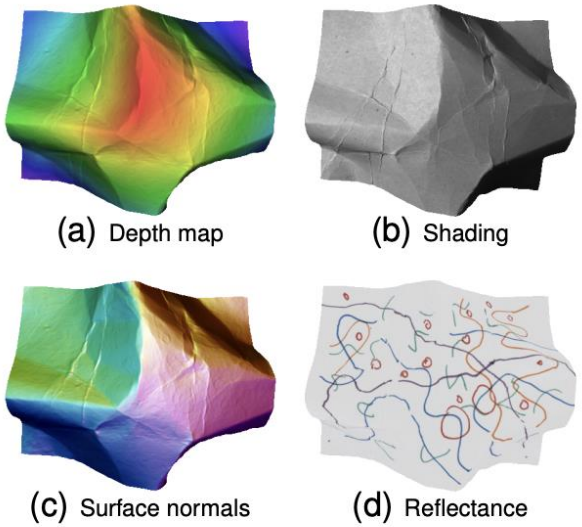
(e) gaffer's explanation

Shape, illumination, and reflectance from shading (TPAMI'15)

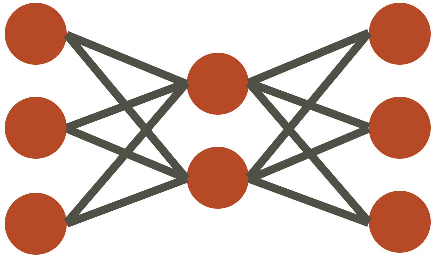
Strategies



Increase observation



Introduce scene priors

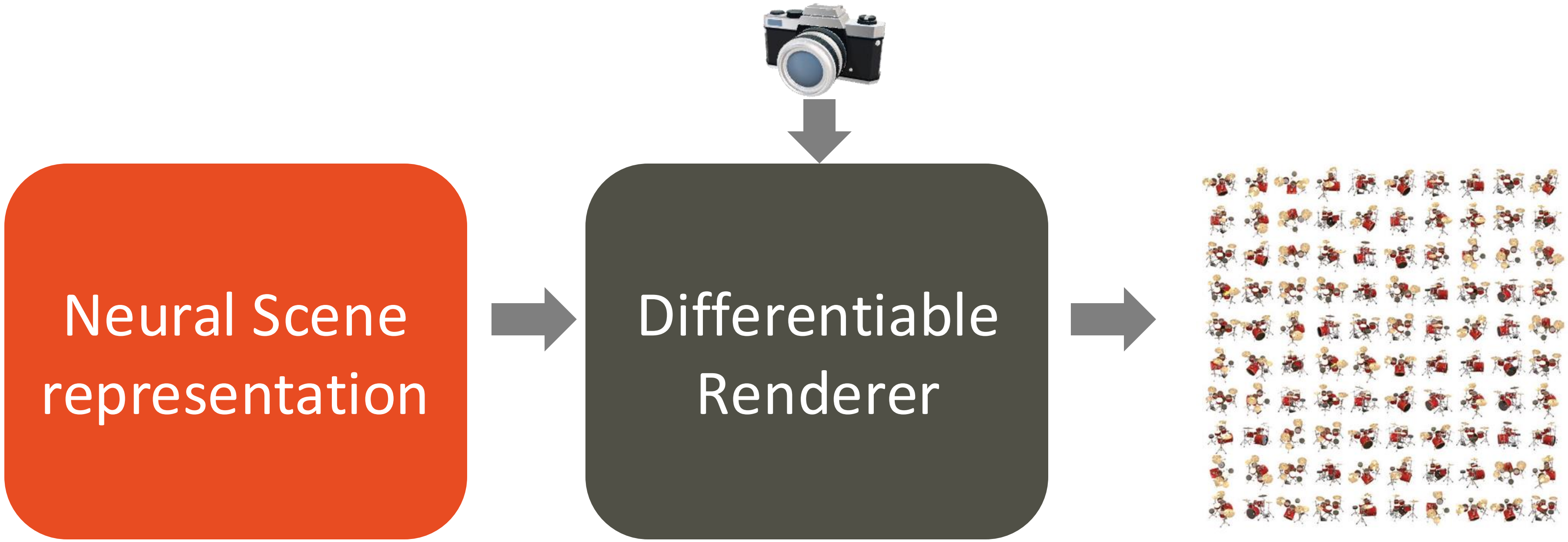


Differentiable rendering

ShadowNeuS: Neural SDF Reconstruction by Shadow Ray Supervision (CVPR'23)
Shape, illumination, and reflectance from shading (TPAMI'15)

Differentiable rendering

1. Set the scene as learnable parameters
2. Differentiable forward rendering



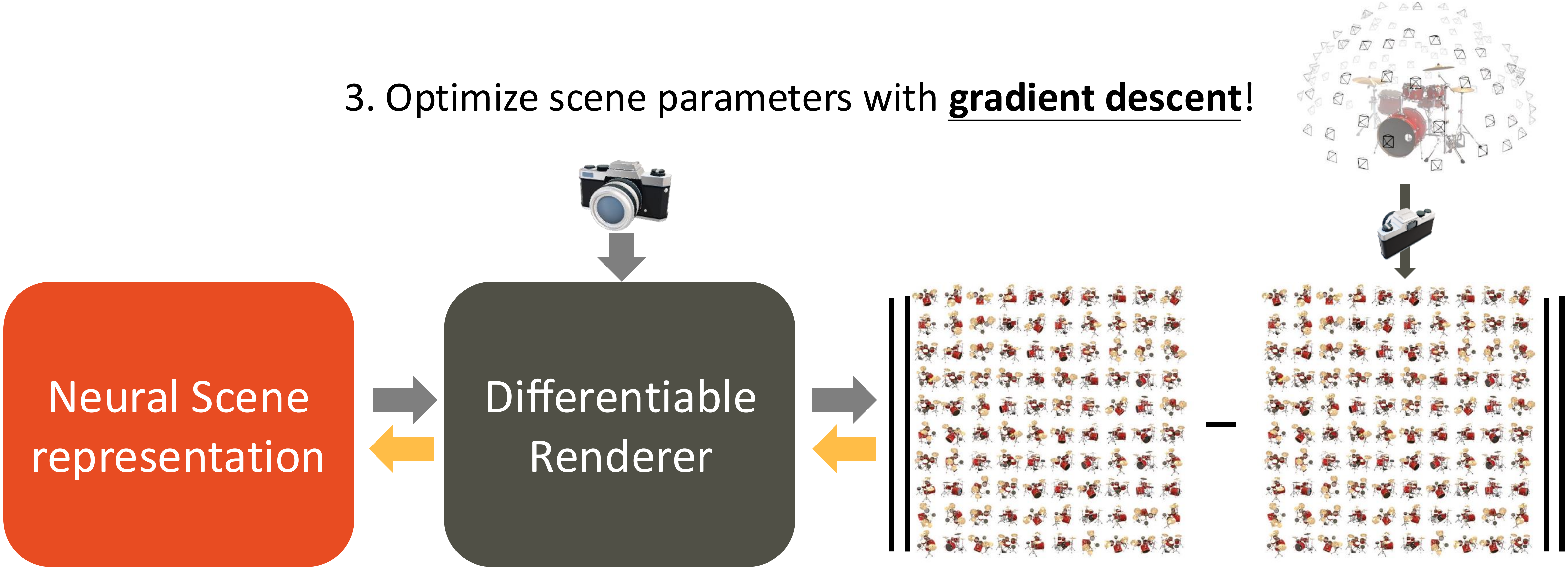
Neural Scene representation

Differentiable Renderer

Rendered images

Differentiable rendering

3. Optimize scene parameters with **gradient descent**!



Neural Scene representation

Differentiable Renderer

Rendered images

GT images

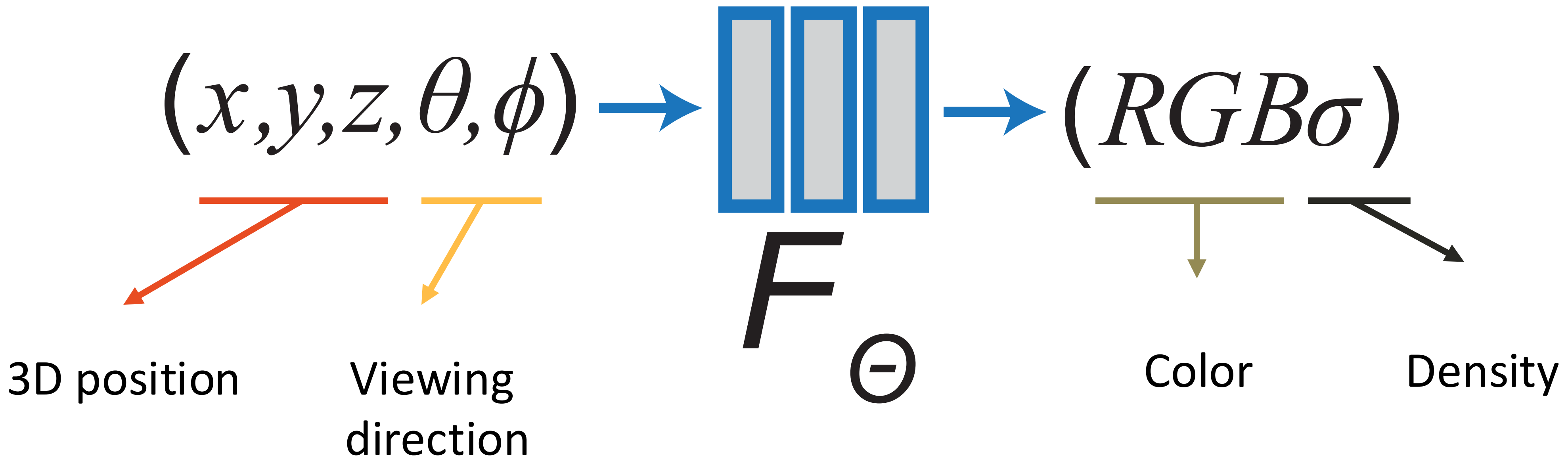
Neural Radiance Field (NeRF)

Multi-view images +
Camera pose

3D scene

NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV'20)

NeRF: representation



$(x,y,z,\theta,\phi) \rightarrow F_\Theta \rightarrow (RGB\sigma)$

3D position

Viewing direction

Color

Density

NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV’20)

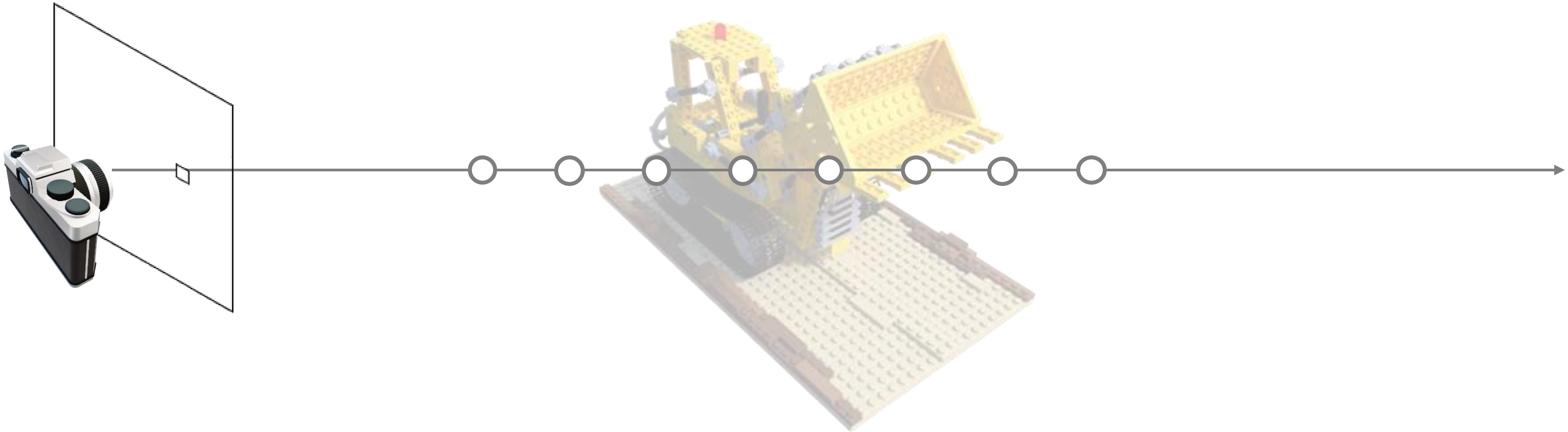
Volume rendering

How to calculate the color of the pixel?

NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV’20)

Volume rendering

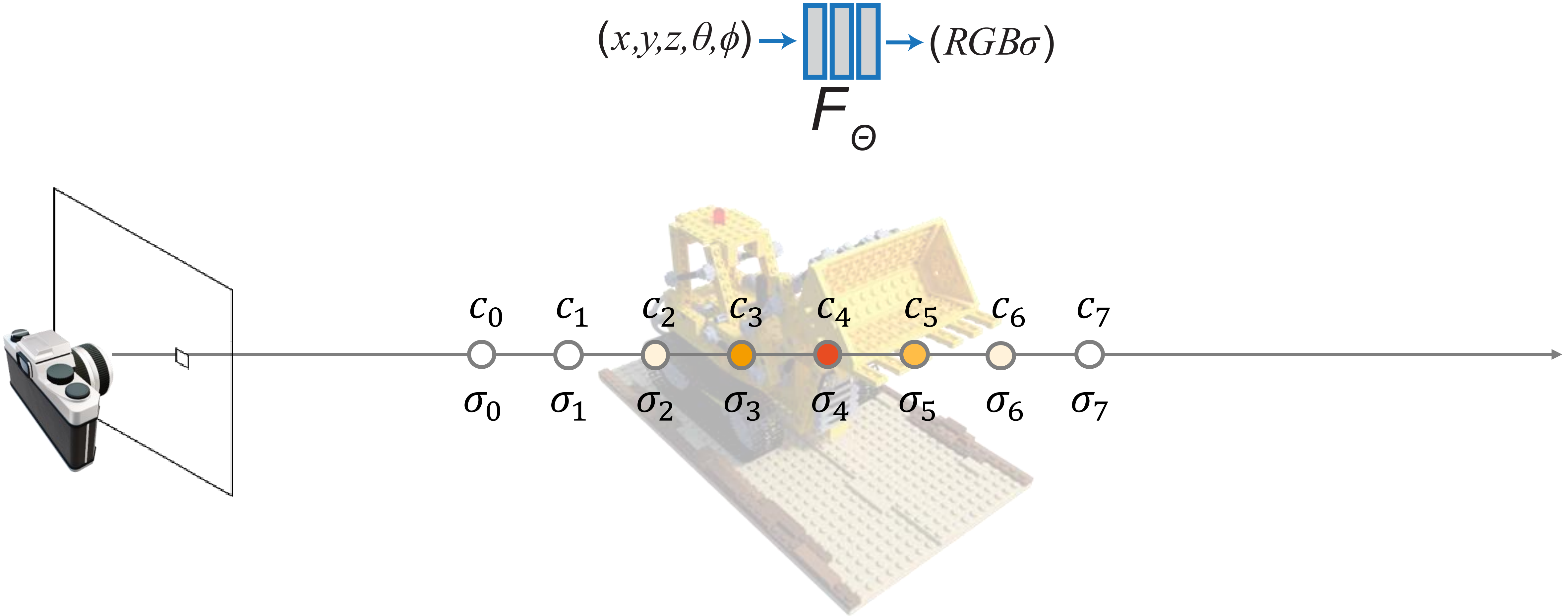
1. Cast a ray from camera to the scene
2. Sample multiple points along the ray



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV’20)

Volume rendering

3. Predict color, density of each point



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV'20)

Volume rendering

Weighted sum of point color

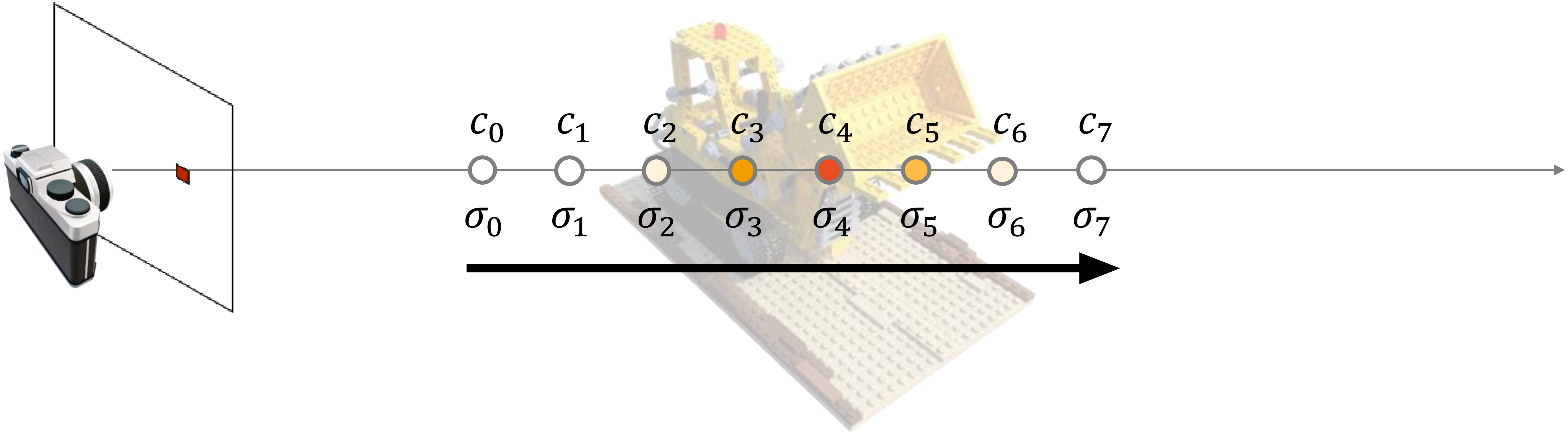
$$C(r) = \sum_{i=1}^{N} T_i \alpha_i c_i$$

How much light is NOT blocked yet

$$T_i = \exp(-\sum_{j=1}^{i-1} \sigma_i \delta_i)$$

How much light is blocked by this point

$$\alpha_i = 1 - \exp(-\sigma_i \delta_i)$$



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV’20)

Volume rendering

Weighted sum of point color

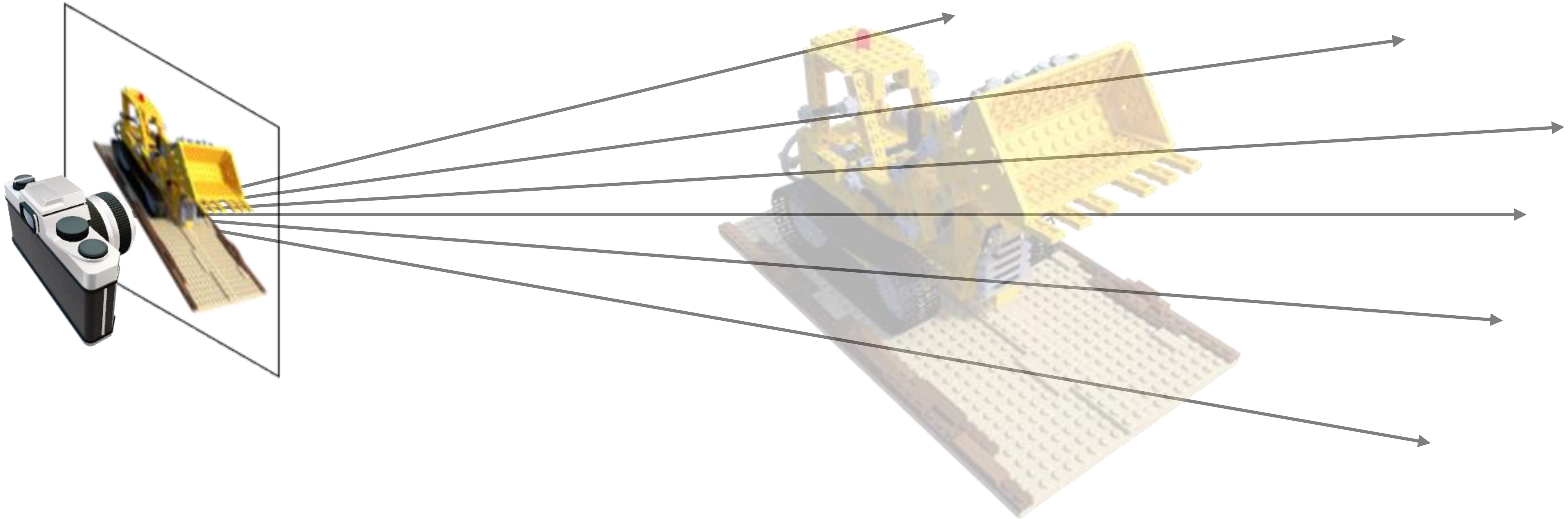
$$C(r) = \sum_{i=1}^{N} T_i \alpha_i c_i$$

How much light is NOT blocked yet

$$T_i = \exp(-\sum_{j=1}^{i-1} \sigma_i \delta_i)$$

How much light is blocked by this point

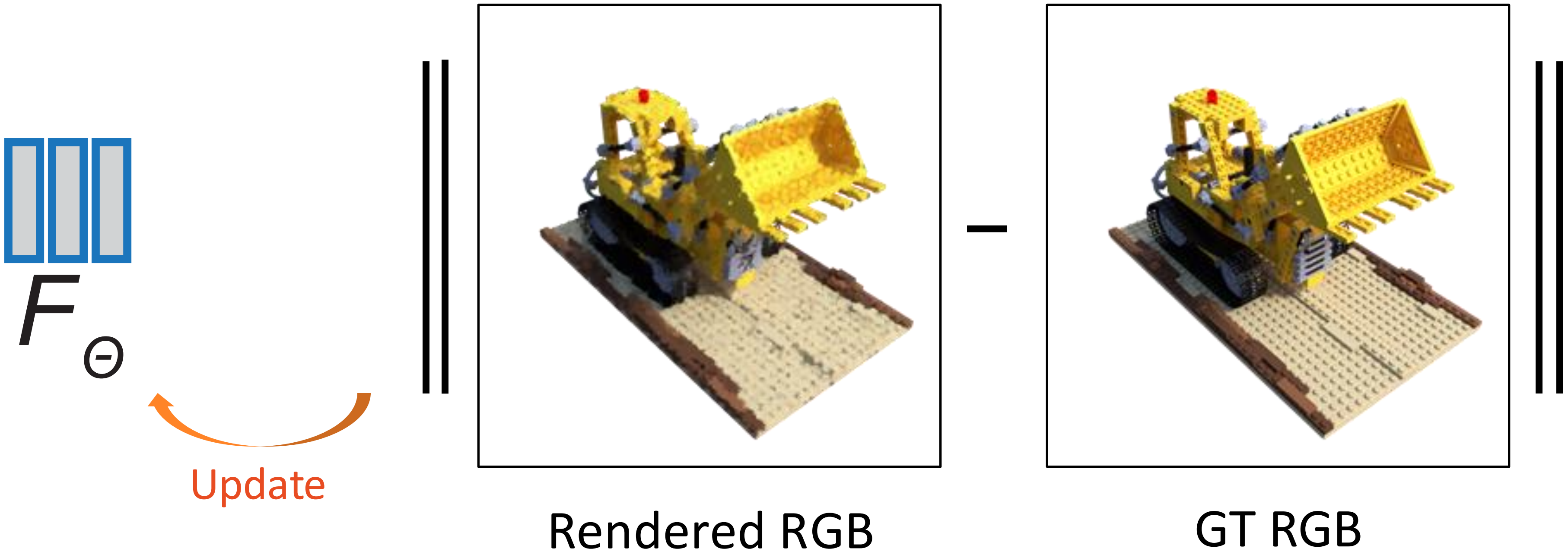
$$\alpha_i = 1 - \exp(-\sigma_i \delta_i)$$



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV'20)

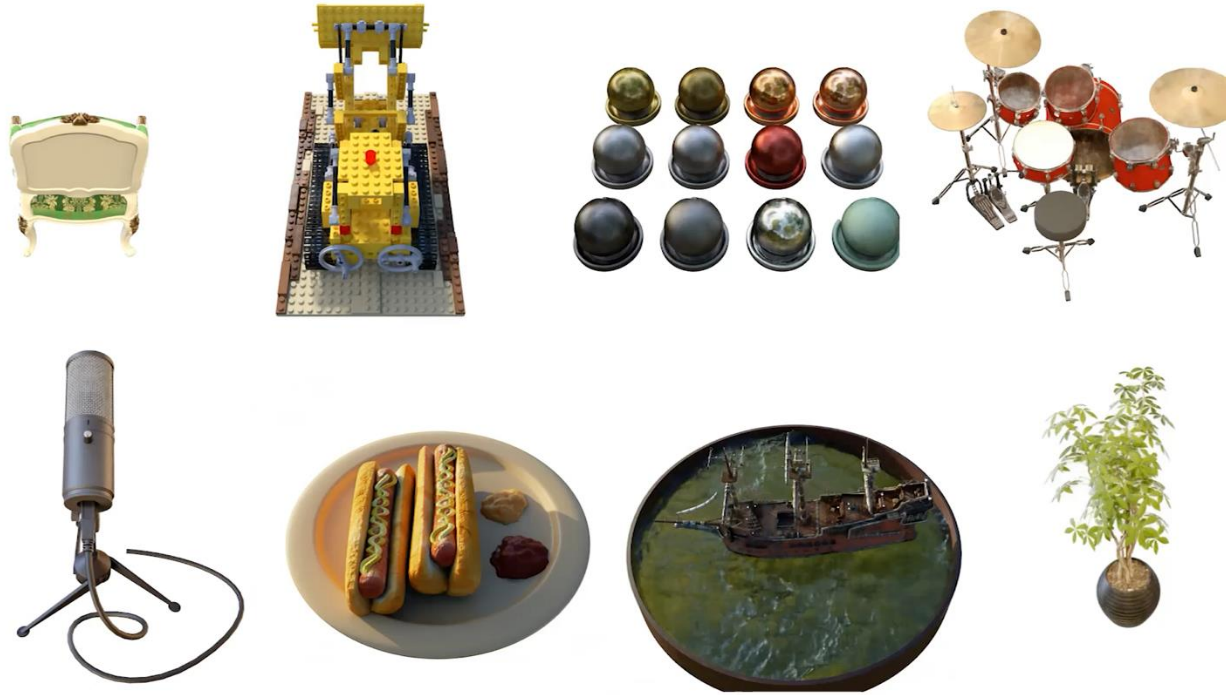
NeRF: optimization

$$L = \sum_{r \in R} ||\overset{\text{Rendered}}{C(r)} - \overset{\text{GT}}{\hat{C}(r)}||_2^2$$



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV'20)

Neural Radiance Field (NeRF)



Strength

- Realistic novel view synthesis
- GT 3D data is not required

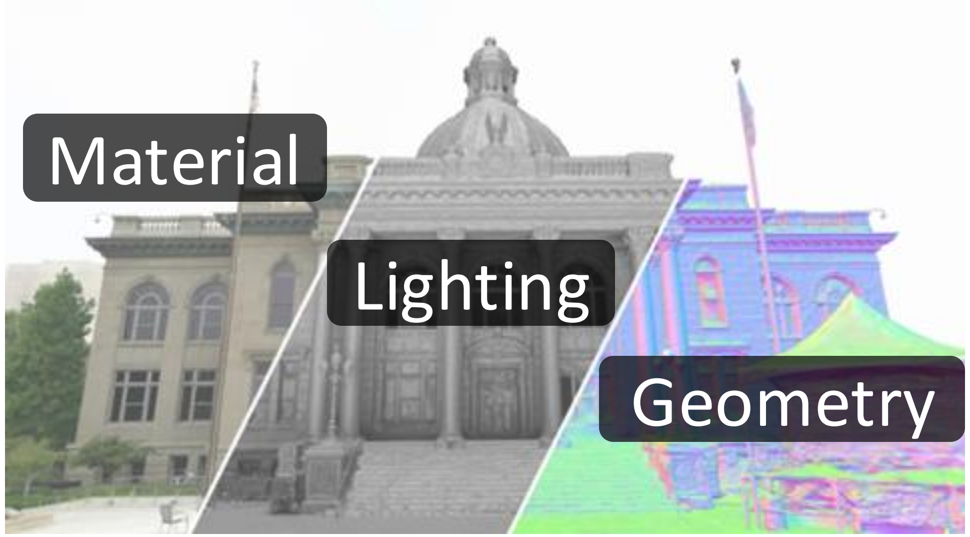
Weakness

- Limited to small scenes
- No lighting and material decomposition
- Slow optimization & rendering

NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis (ECCV’20)

We have talked about …

- What are the basic components of rendering process?
  - Geometry, lighting, material, BRDF
- How to render an image?
  - Rendering equation, Ray tracing, denoising
- What is inverse rendering? Why is it challenging?
  - Ill-posed problem, incomplete observation, limited data



3D scene

Render

Reconstruct

2D images/videos

We have NOT talked about …

- How to parameterize lighting and material (e.g. BRDF)?
- What are other ways to render an image?
    - Rasterization, 3D Gaussian Splatting
- How to estimate smooth surfaces from images?
- How to estimate lighting and material?
- How to insert virtual objects in real-world images/videos?
- Data-driven approaches for inverse rendering
- …

We will discuss some of the topics in the following lectures!