CVSE Presentation on Application of NERF in Astrophysics

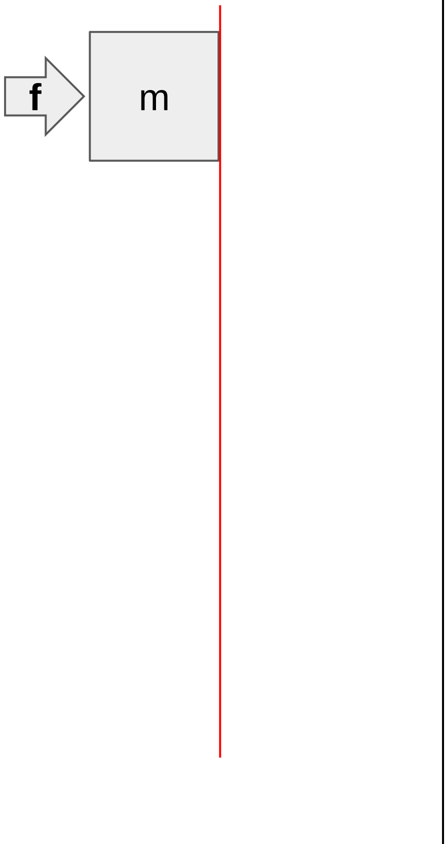
Xu Zhang (New York University)
xz4863@nyu.edu

Context

Context

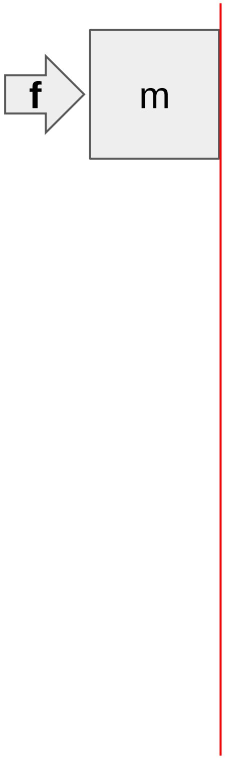
Sunspot Numbers and Radiance Fields

Small force
Light object

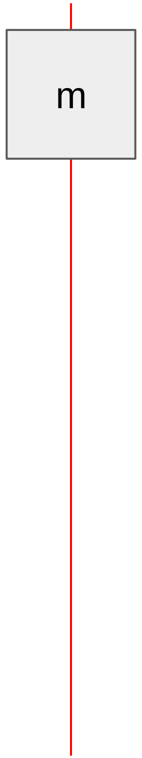


t=0

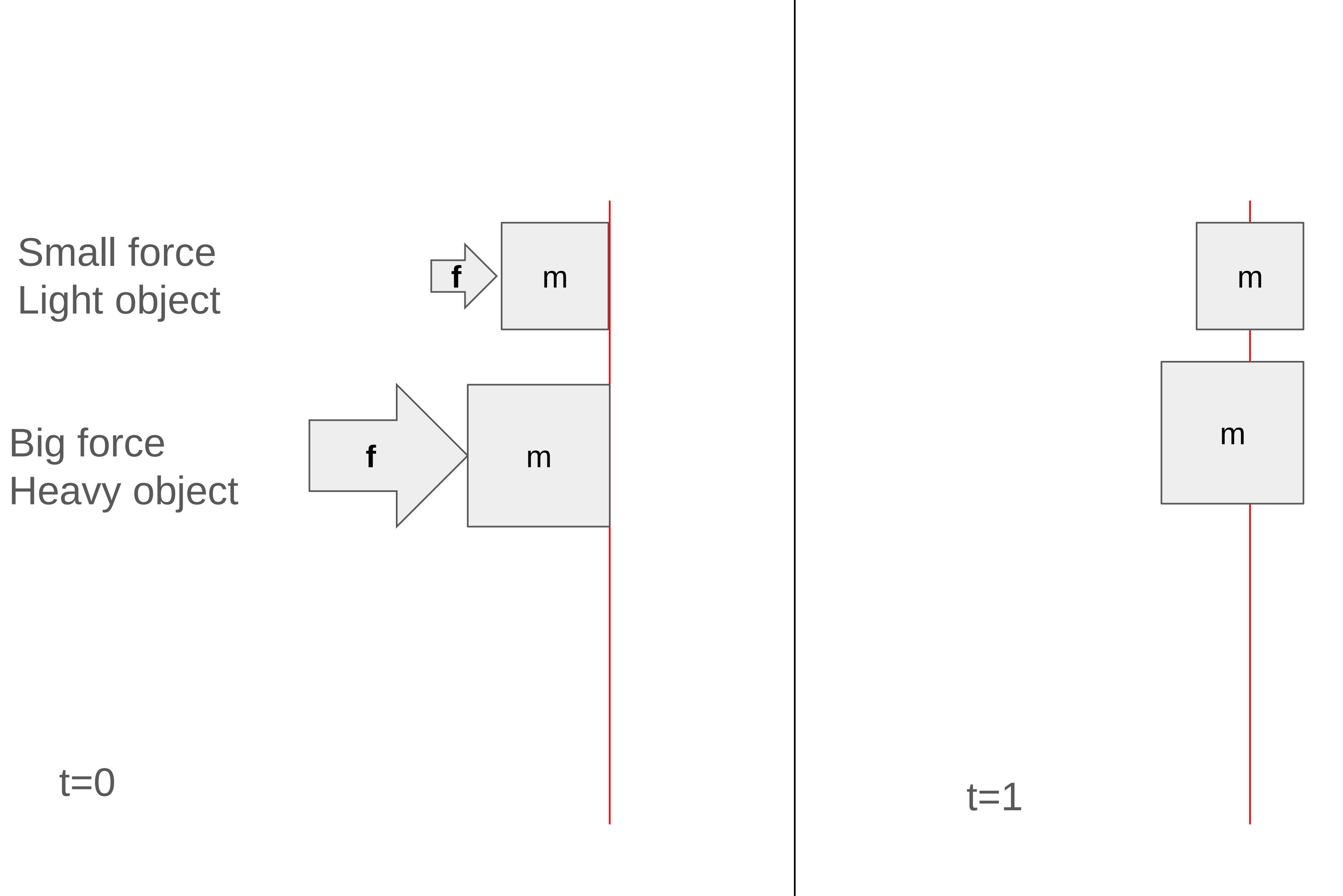
Small force
Light object

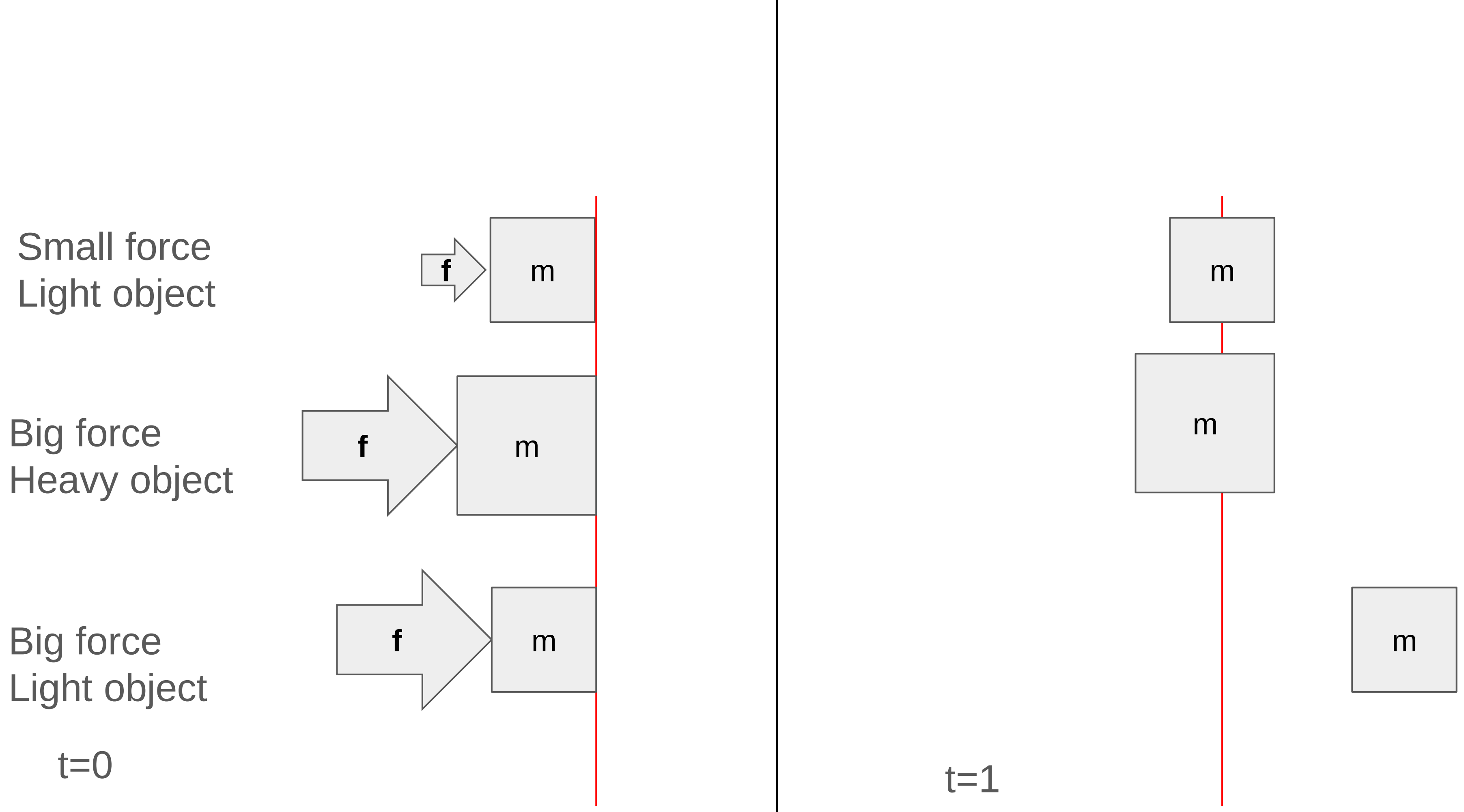


t=0



t=1


Small force
Light object
f
m
Big force
Heavy object
f
m
t=0
m
m
t=1

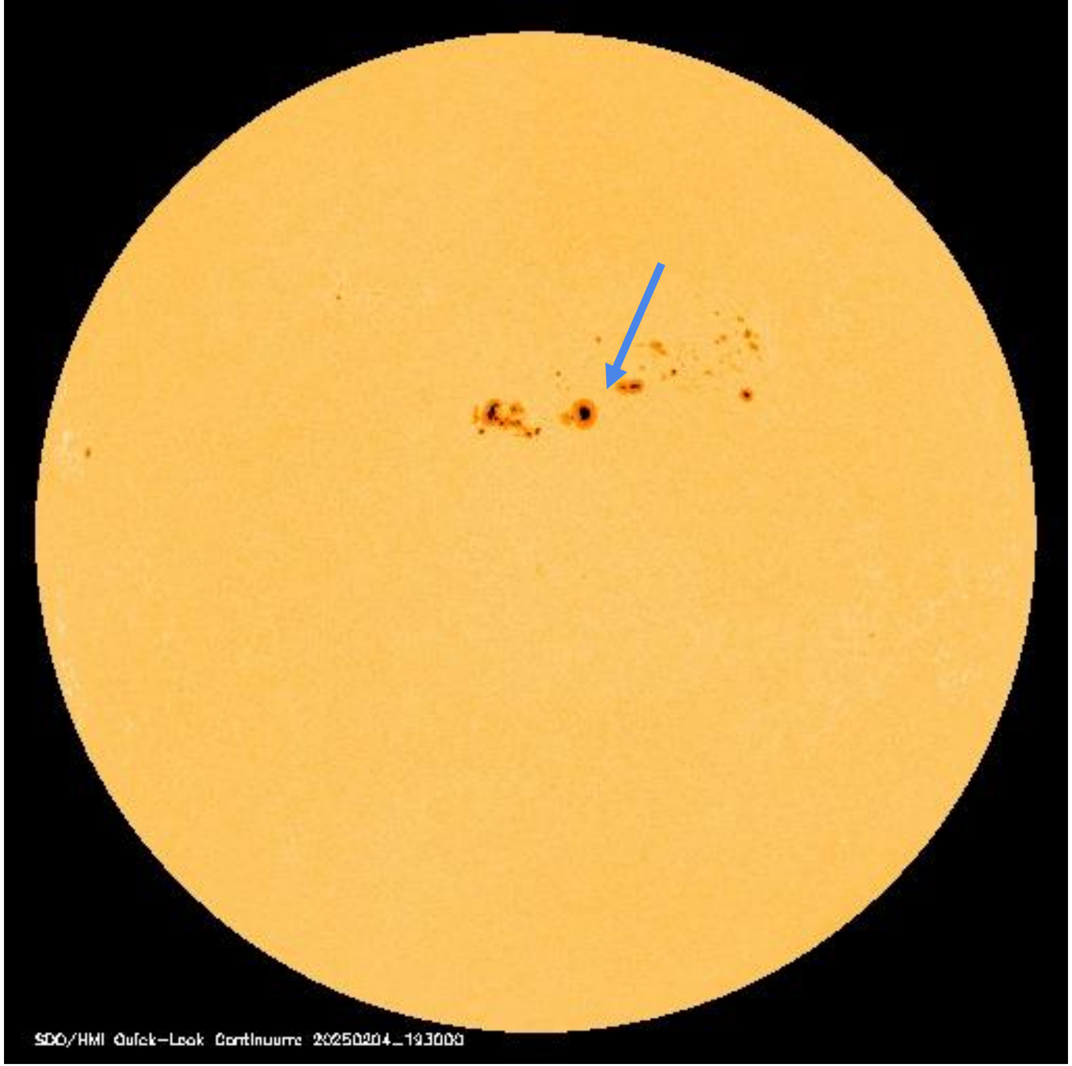

Small force
Light object
f
m
Big force
Heavy object
f
m
Big force
Light object
f
m
t=0
m
m
m
t=1

Studying Astrophysics

Standard image of the Sun taken from space Monday Feb 4, 2025

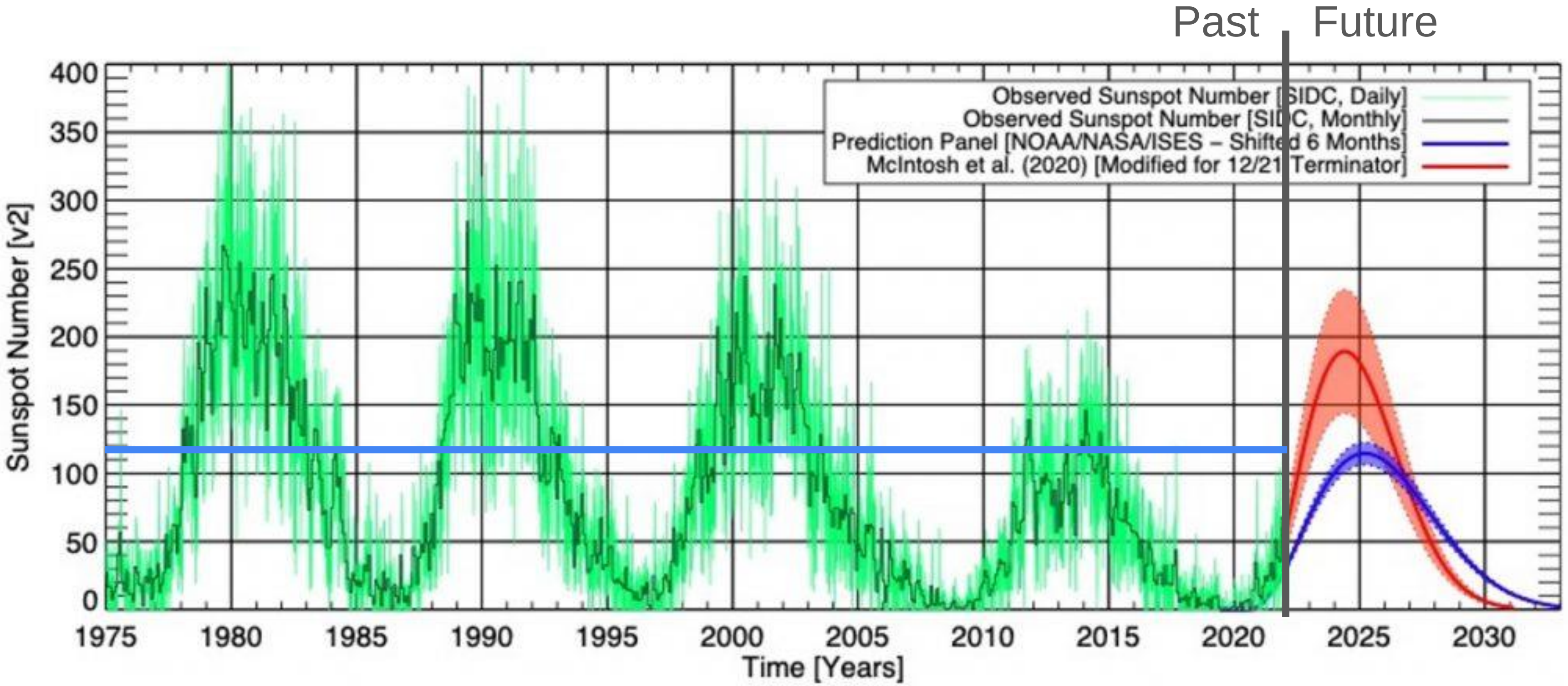
"Active Region"/Sunspot that has strong magnetic field and is cooler (and hence darker)

Active regions are related to flares and other space weather



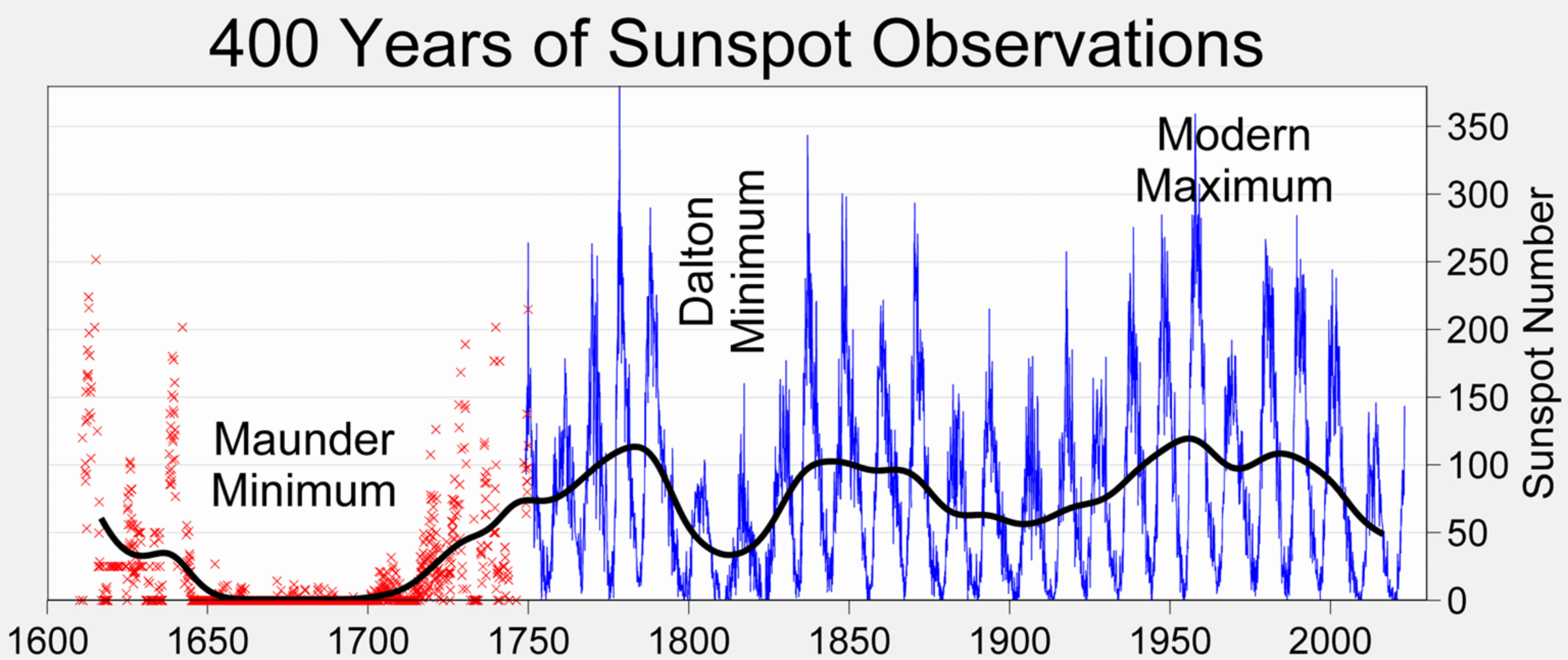
Source: https://soho.nascom.nasa.gov/data/realtime/hmi_igr/512/

Sunspots Over Time



Source: https://blogs.nasa.gov/solarcycle25/2022/07/27/solar-cycle-25-is-exceeding-predictions-and-showing-why-we-need-the-gdc-mission/

Going Further Back…



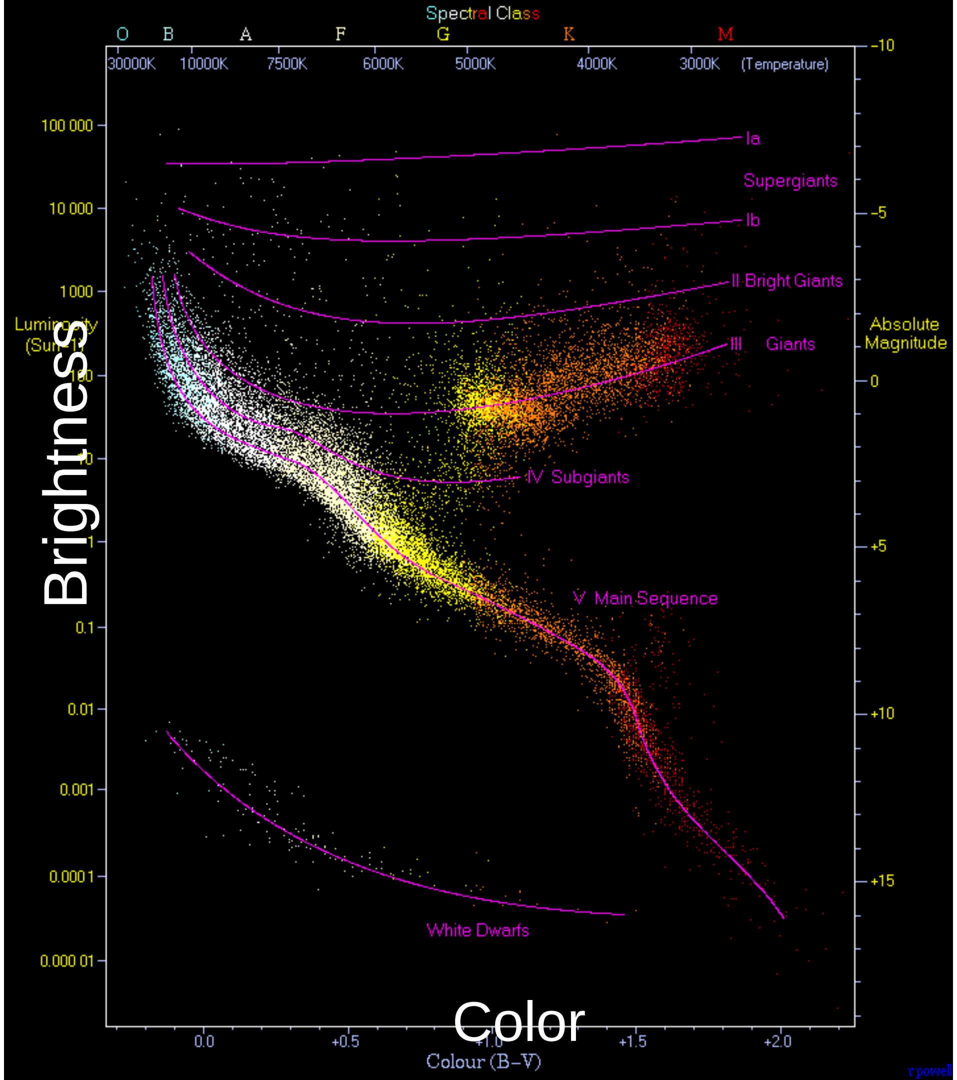
Source: https://en.wikipedia.org/wiki/Solar_cycle#/media/File:Sunspot_Numbers.png

How?

- Want to learn how universe works. Can’t use just one sample
- Want to look at *lots* of things

How?

- Want to learn how universe works. Can’t use just one sample
- Want to look at *lots* of things

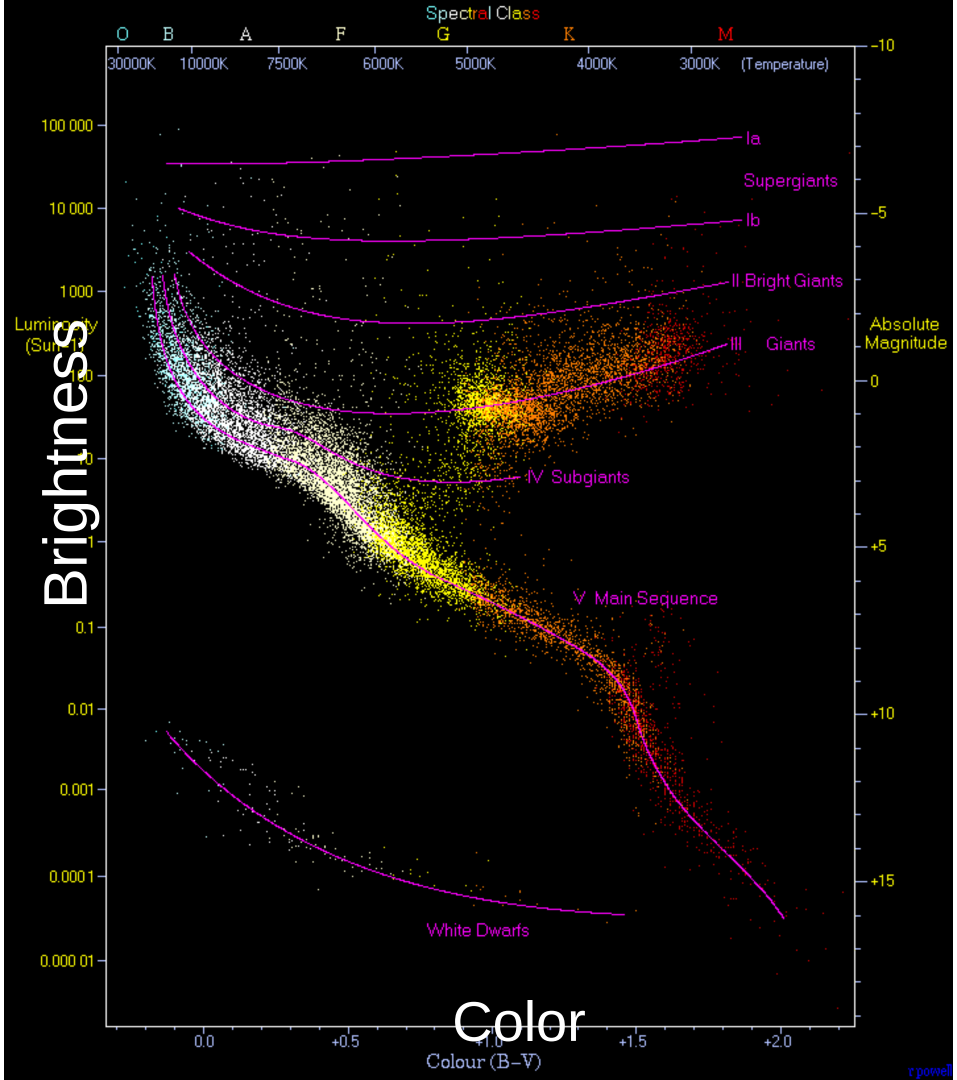


Source: https://en.wikipedia.org/wiki/Hertzsprung%E2%80%93Russell_diagram

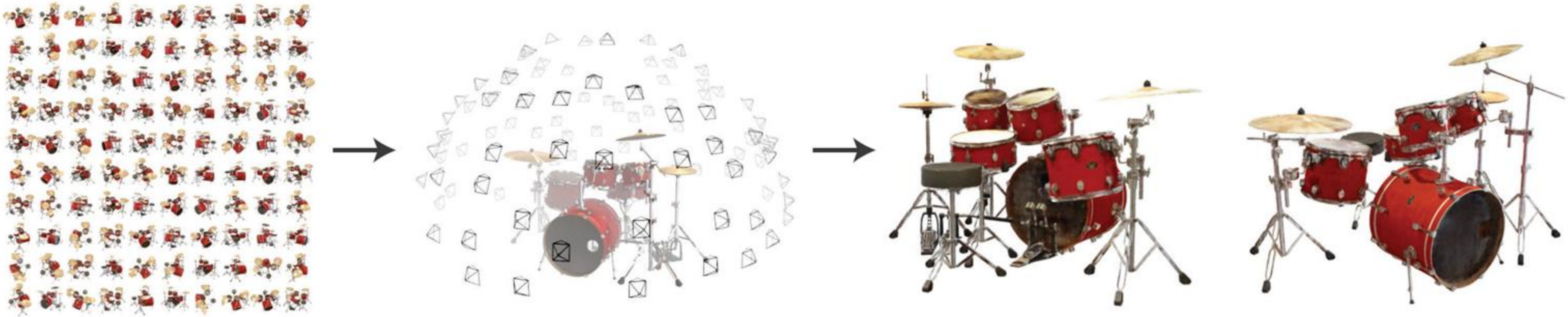
How?

- Want to learn how universe works. Can't use just one sample
- Want to look at *lots* of things
- Don't want to do it manually and often can't do it manually

Source: https://en.wikipedia.org/wiki/Hertzsprung%E2%80%93Russell_diagram



Modeling Scenes



Goal: take N views of a scene with poses, produce new views

Photo credit: Mildenhal et al. ECCV 2022

Modeling Scenes – Rendering Equation

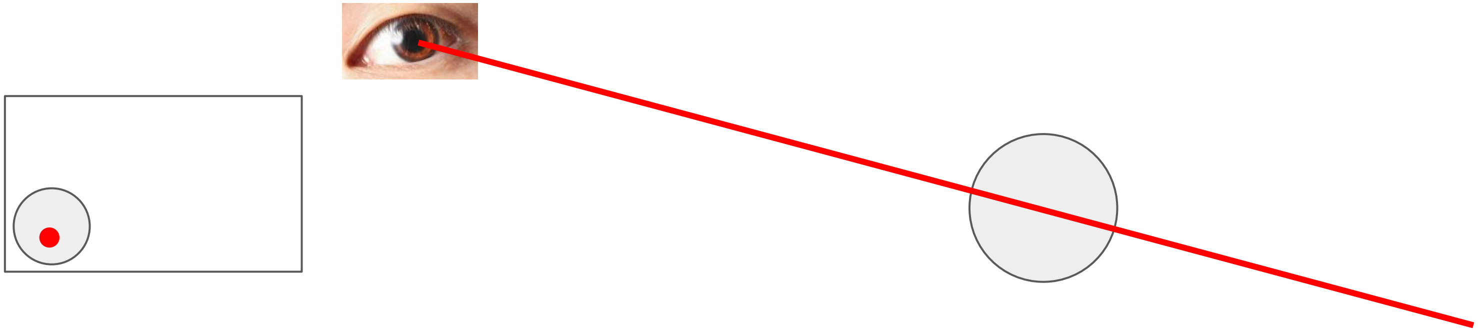
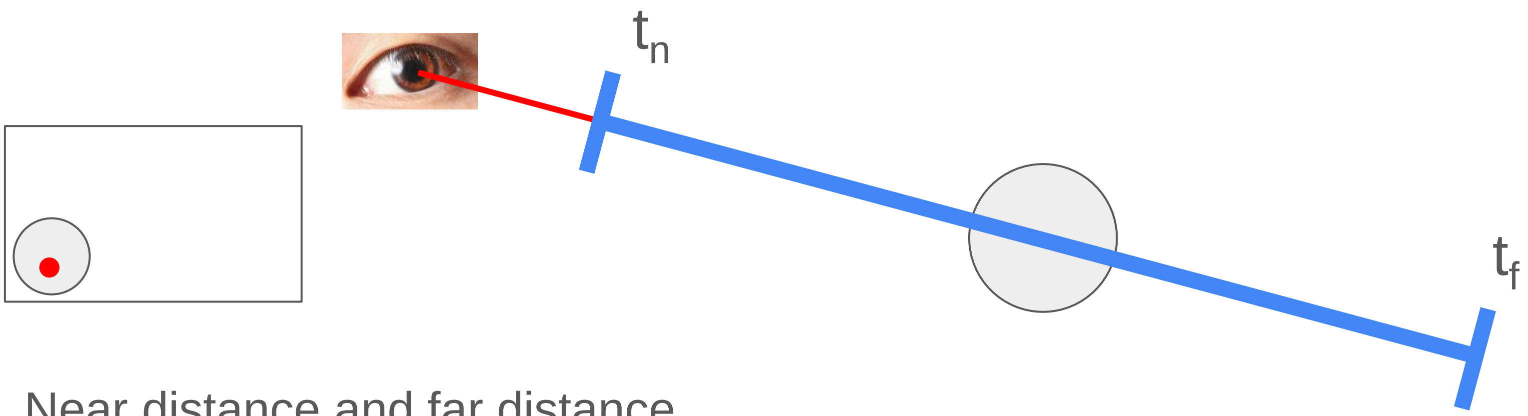


Photo credit: Wikipedia, *Human Eye*

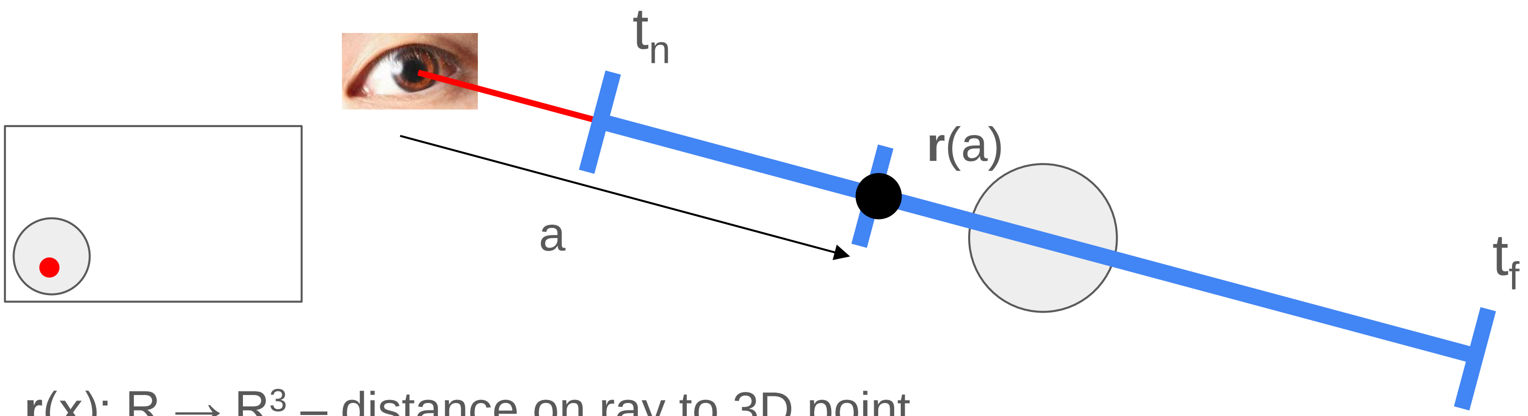
Modeling Scenes – Rendering Equation



Near distance and far distance

Photo credit: Wikipedia, *Human Eye*

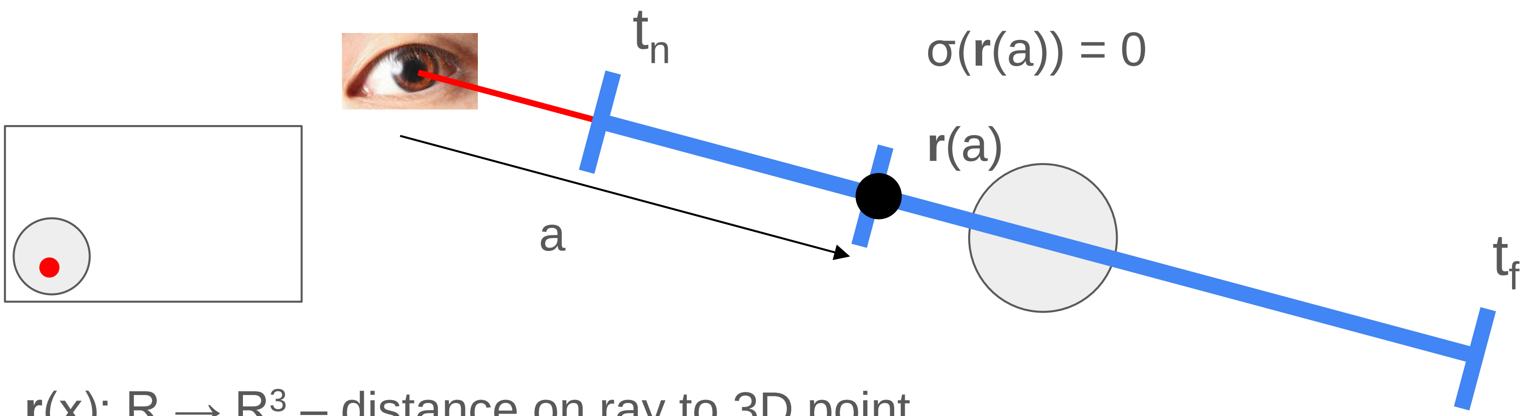
Modeling Scenes – Rendering Equation



$\mathbf{r}(x)$: $R \rightarrow R^3$ – distance on ray to 3D point

Photo credit: Wikipedia, *Human Eye*

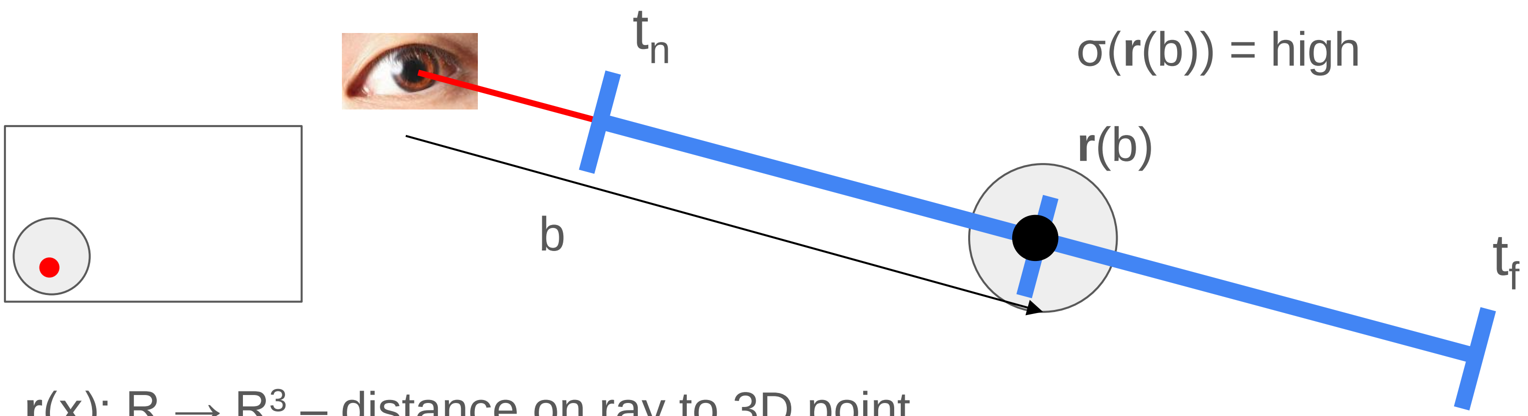
Modeling Scenes – Rendering Equation



$\mathbf{r}(x)$: $R \rightarrow R^3$ – distance on ray to 3D point
$\sigma(\mathbf{p})$: $R^3 \rightarrow R$ – density at a 3D point

Photo credit: Wikipedia, *Human Eye*

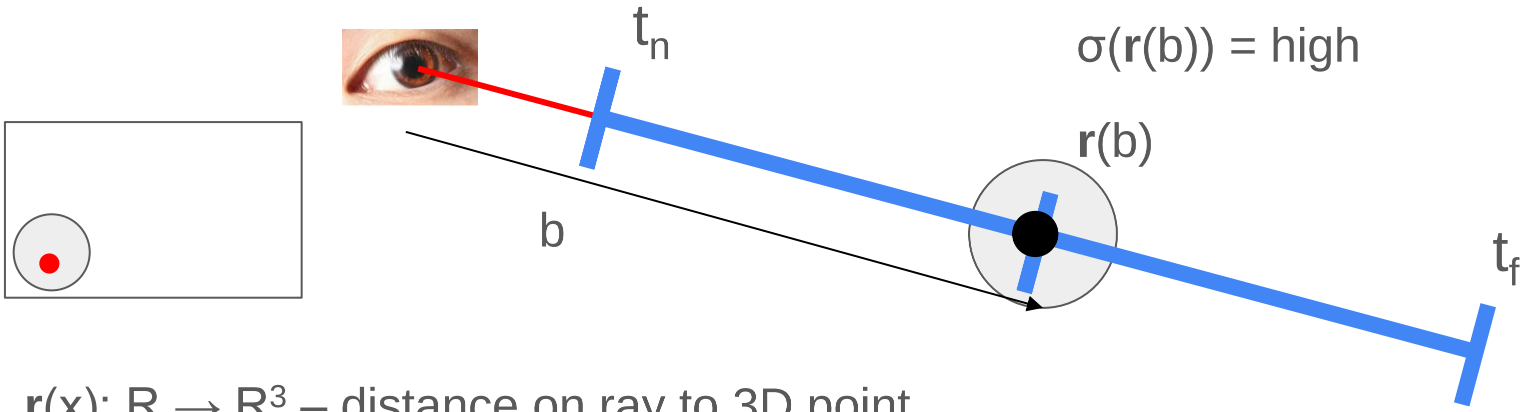
Modeling Scenes – Rendering Equation



$\mathbf{r}(x)$: $R \rightarrow R^3$ – distance on ray to 3D point
$\sigma(\mathbf{p})$: $R^3 \rightarrow R$ – density at a 3D point

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – Rendering Equation

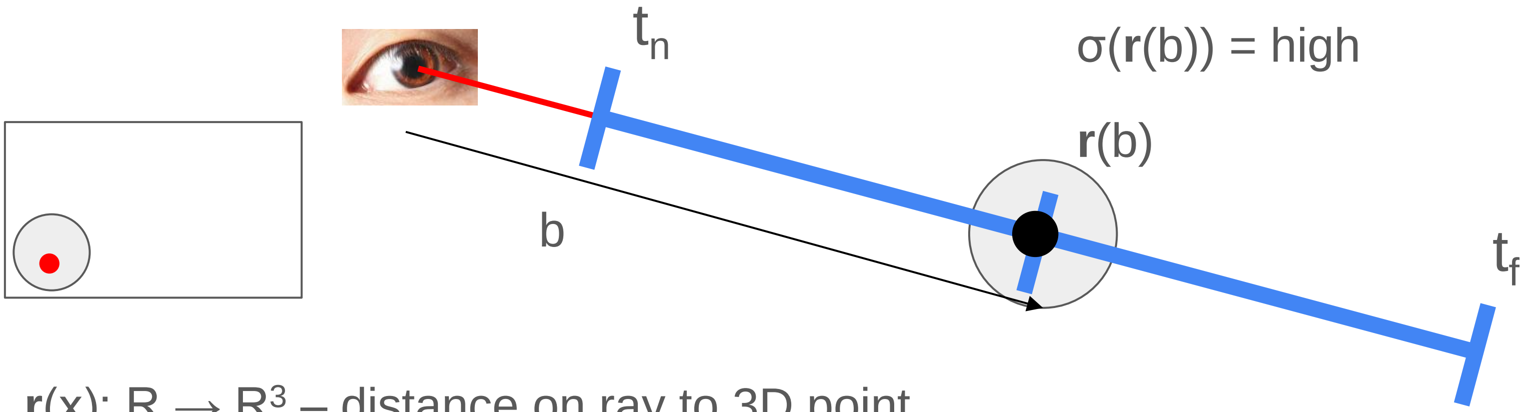


$\mathbf{r}(x)$: $R \rightarrow R^3$ – distance on ray to 3D point
$\sigma(\mathbf{p})$: $R^3 \rightarrow R$ – density at a 3D point

$$T(t) = \exp\left(-\int_{t_n}^{t} \boxed{\sigma(\mathbf{r}(s))} ds\right)$$ Density s units away

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – Rendering Equation

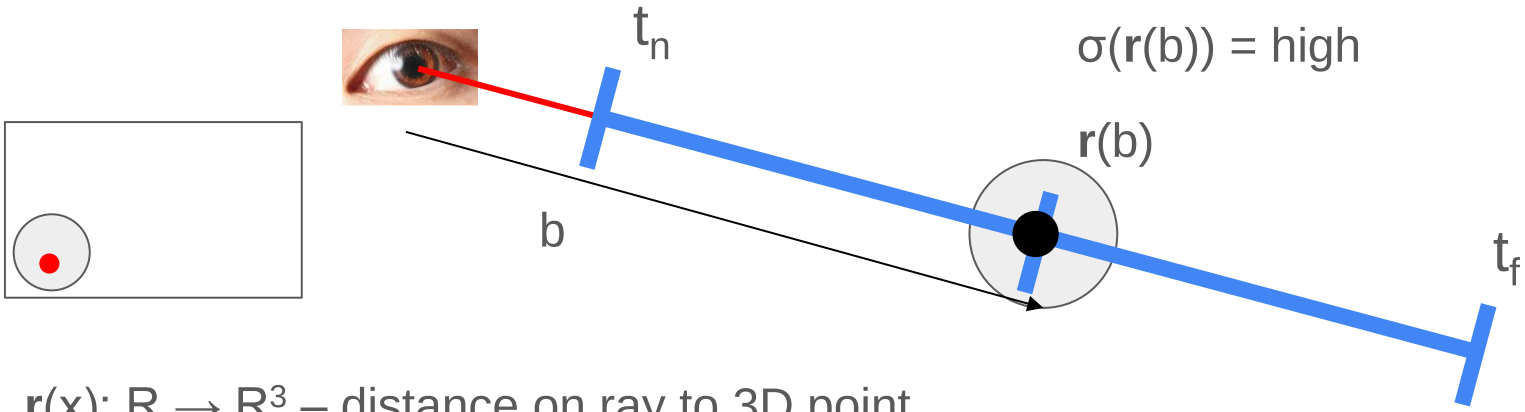


$\mathbf{r}(x)$: $R \rightarrow R^3$ – distance on ray to 3D point
$\sigma(\mathbf{p})$: $R^3 \rightarrow R$ – density at a 3D point

$$T(t) = \exp\left(-\boxed{\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds}\right)$$ Total density from t_n (near) to location

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – Rendering Equation



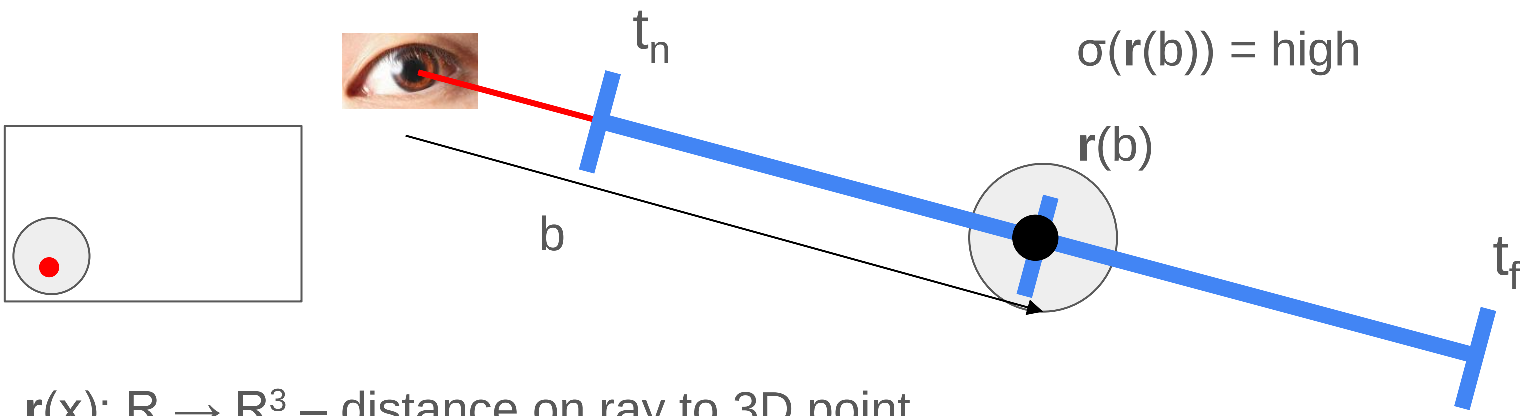
$\mathbf{r}(x)$: $R \rightarrow R^3$ – distance on ray to 3D point
$\sigma(\mathbf{p})$: $R^3 \rightarrow R$ – density at a 3D point

$$T(t) = \boxed{\exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right)}$$

If total density up to point t is 0: 1
If total density up to point t is high: 0

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – Rendering Equation



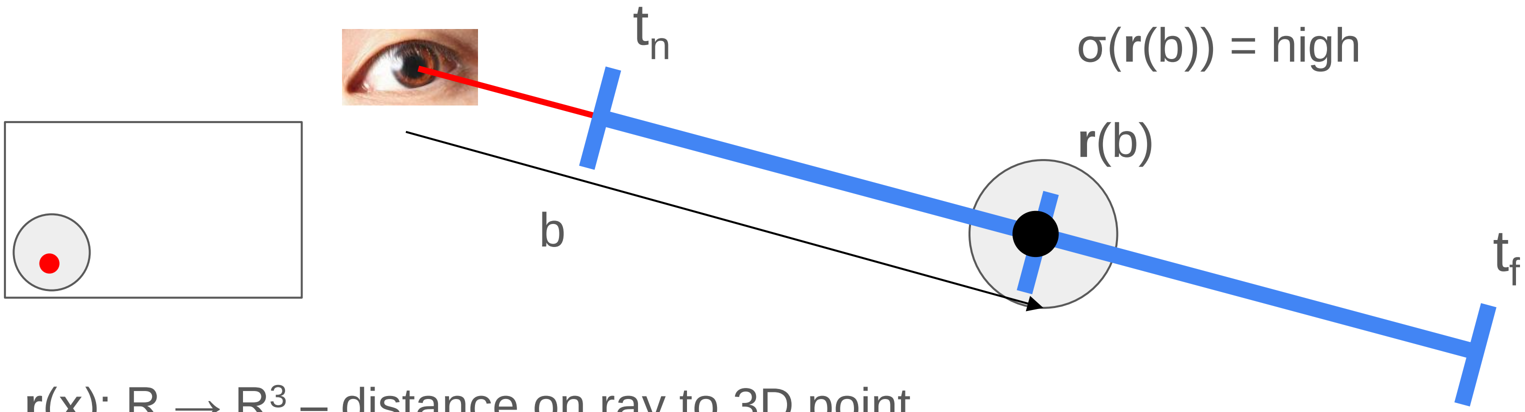
r(x): R → R^3 – distance on ray to 3D point
σ(**p**): R^3→ R – density at a 3D point
T(t): R → R – transmittance of a location on the ray

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\boxed{\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t)\quad)}dt$$

Color multiplied by density t units along the ray

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – Rendering Equation



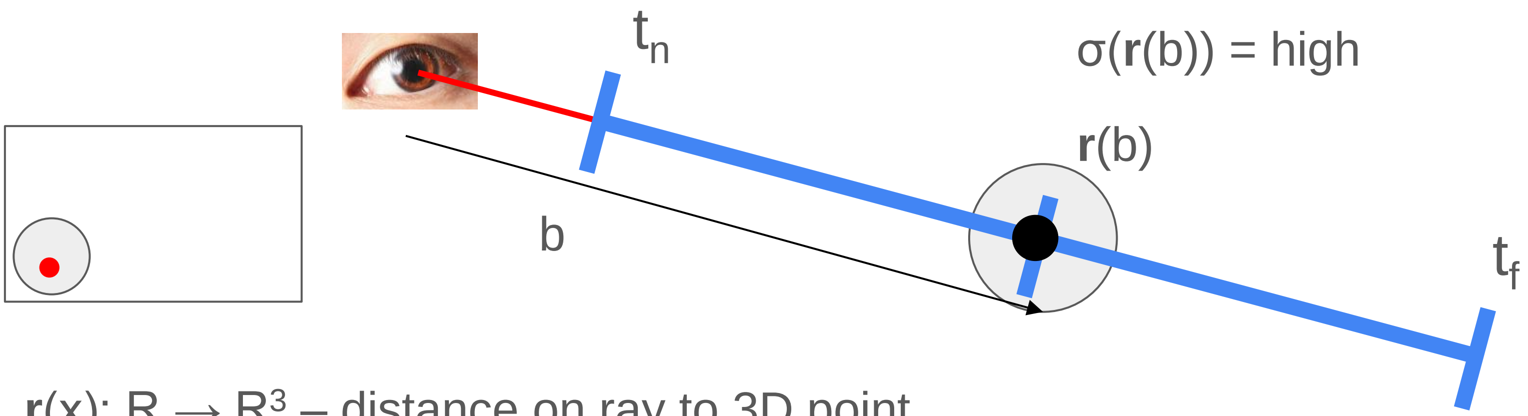
$\mathbf{r}(x)$: R → R^3 – distance on ray to 3D point
$\sigma(\mathbf{p})$: R^3→ R – density at a 3D point
T(t): R → R – transmittance of a location on the ray

$$C(\mathbf{r}) = \int_{t_n}^{t_f} \boxed{T(t)}\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t) \qquad)\boxed{dt}$$

Weighted by the transmittance.
Anything dense in front = not seen

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – Rendering Equation



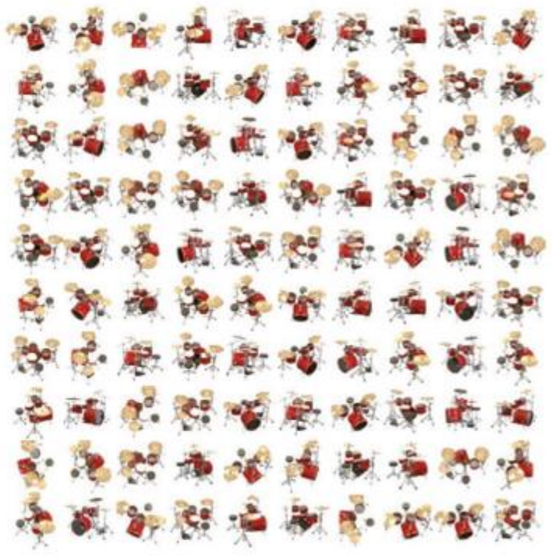
$\mathbf{r}(x)$: R → R^3 – distance on ray to 3D point
$\sigma(\mathbf{p})$: R^3→ R – density at a 3D point
T(t): R → R – transmittance of a location on th eray

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt$$

Full version includes viewing direction
~Correct formula for normal conditions

Photo credit: Wikipedia, *Human Eye*

Modeling Scenes – One Method



Option 1: create *giant* 5D look-up table

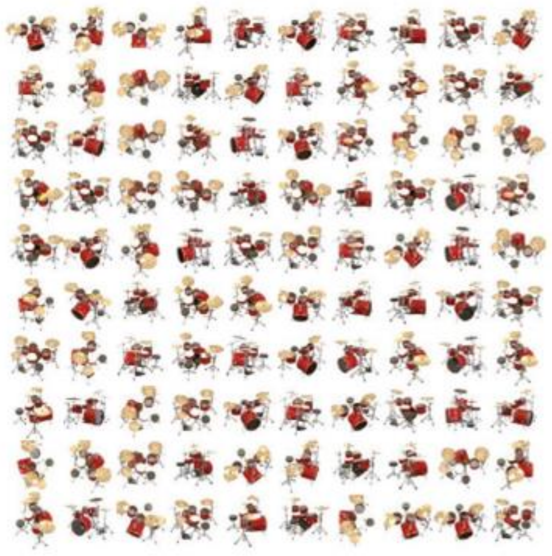
$V[x,y,z,\theta,\varphi] \rightarrow R,G,B, \sigma$

Downsides:

- Huge memory scaling
- Doesn’t interpolate

Credit for this way of explaining: Andrew Owens, EECS 442; Mildenhal et al. ECCV 2022

Modeling Scenes – One Method



Option 2: learn *giant* 5D look-up table

$f([x,y,z,\theta,\varphi]) \rightarrow R,G,B,\sigma$

Upsides:

- Compresses well
- Interpolates

Credit for this way of explaining: Andrew Owens, EECS 442; Mildenhal et al. ECCV 2022

Three Important Details

Problem underconstrained: usually need **many** views that see each location [x,y,z]
(plus lots of time to fit the model if tricks aren't used)

$$C(\mathbf{r}) = \boxed{\int_{t_n}^{t_f}} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})\boxed{dt}$$

← Don't do actual integral
Either sum (or more tricks)

$$f([\boxed{x,y,z},\theta,\varphi]) \rightarrow R,G,B,\sigma$$

← Don't use raw coordinates
Convert to higher dimensional space

NERF + Black Hole Knowledge

Gravitationally Lensed Black Hole Emission Tomography

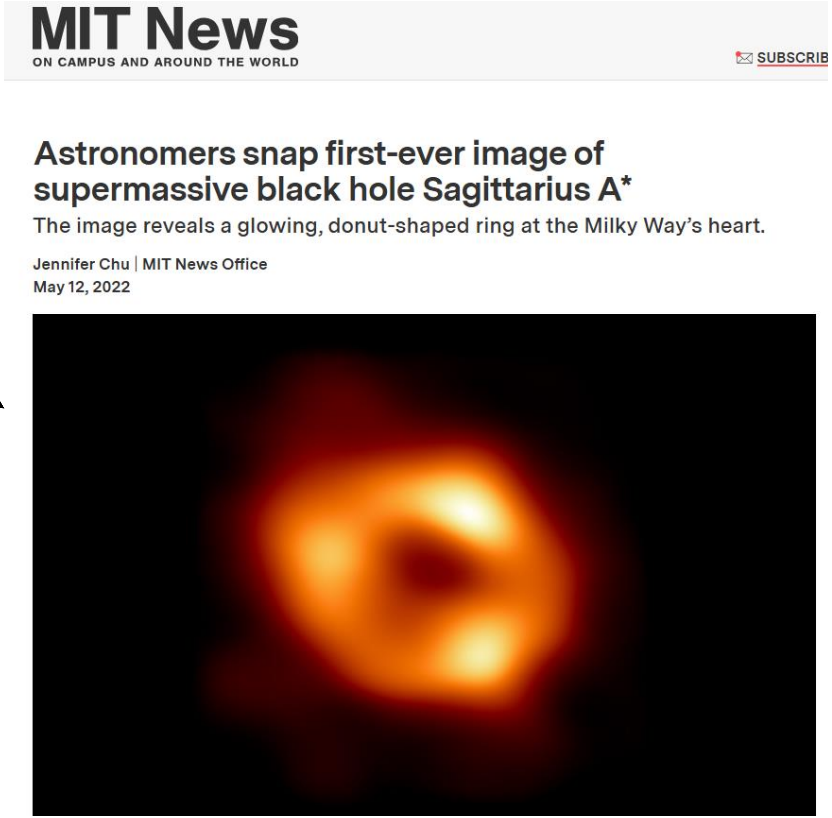
Aviad Levis, et.al

Source: [1] Levis, Aviad, et al. "Gravitationally lensed black hole emission tomography." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

The outline:

Black Hole Reconstruction (Tomography)

You might see on news.
Images of the Black Hole
SgrA$*$


MIT News
ON CAMPUS AND AROUND THE WORLD
SUBSCRIB

Astronomers snap first-ever image of supermassive black hole Sagittarius A*

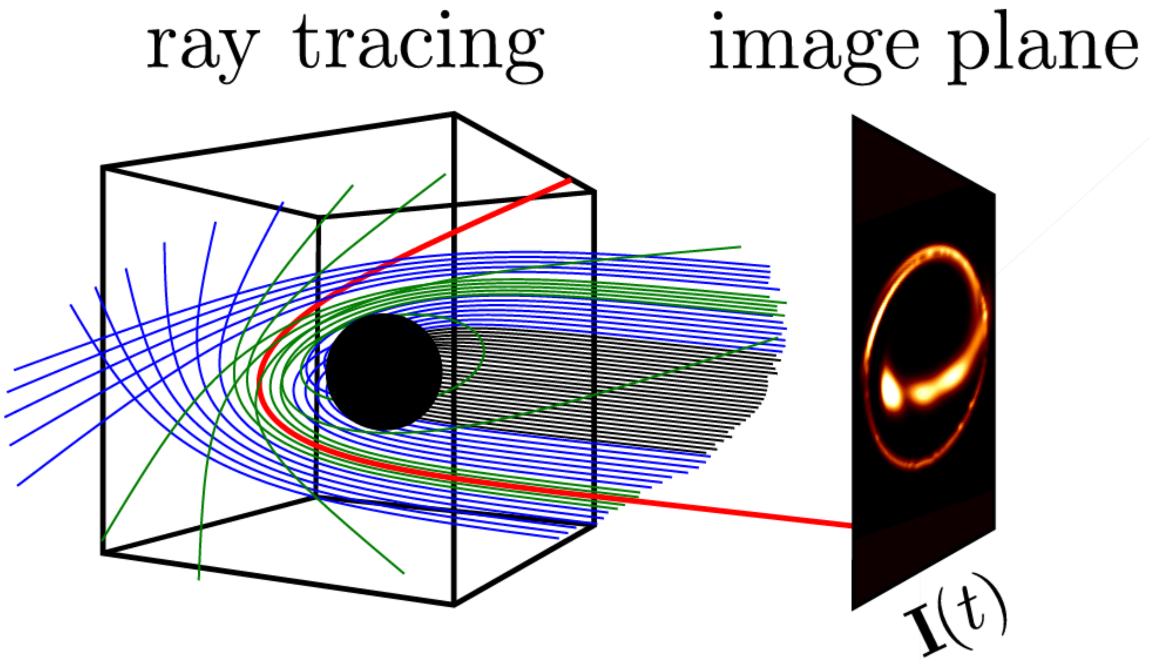
The image reveals a glowing, donut-shaped ring at the Milky Way's heart.

Jennifer Chu | MIT News Office
May 12, 2022

Images are deceiving: Mechanism of black holes

Emission of the black hole at
time t and location **x**
(i.e. **x** is the (*x*,*y*,*z*) coordinate):

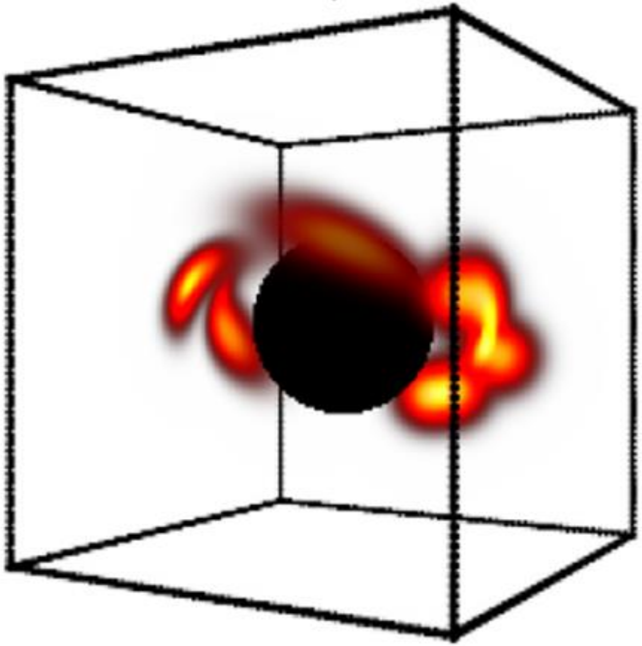
$$e(t, \mathbf{x})$$

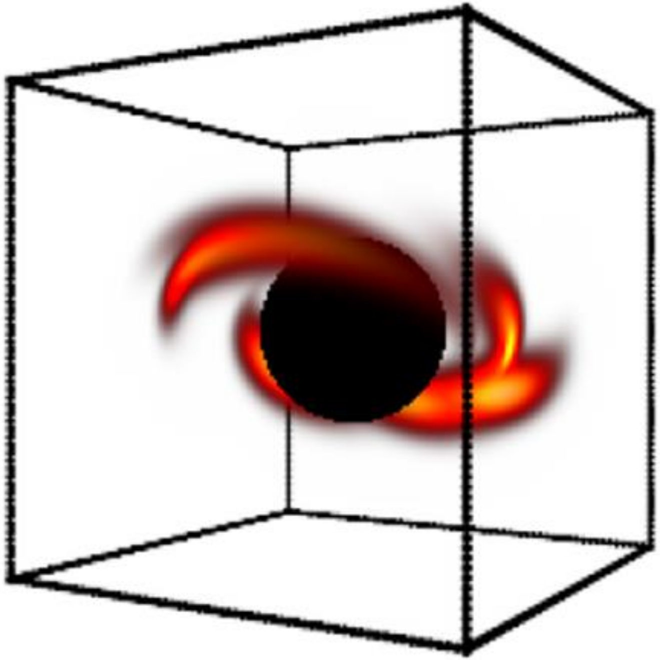


Change of emission from time 0 to time t:

$e_0(x) = e(0, x)$

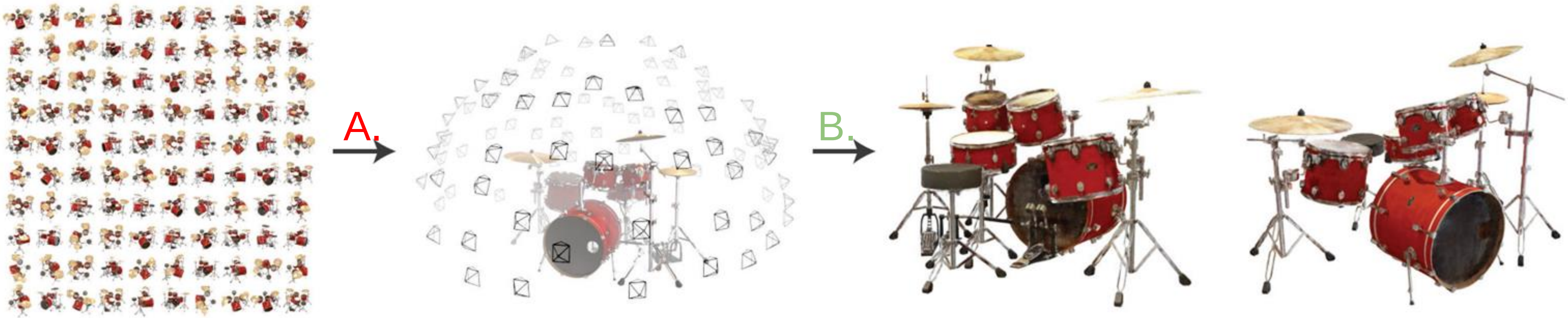
$e(t, \mathbf{x})$





Need a model that can

take 2-D images to create 3-D representations

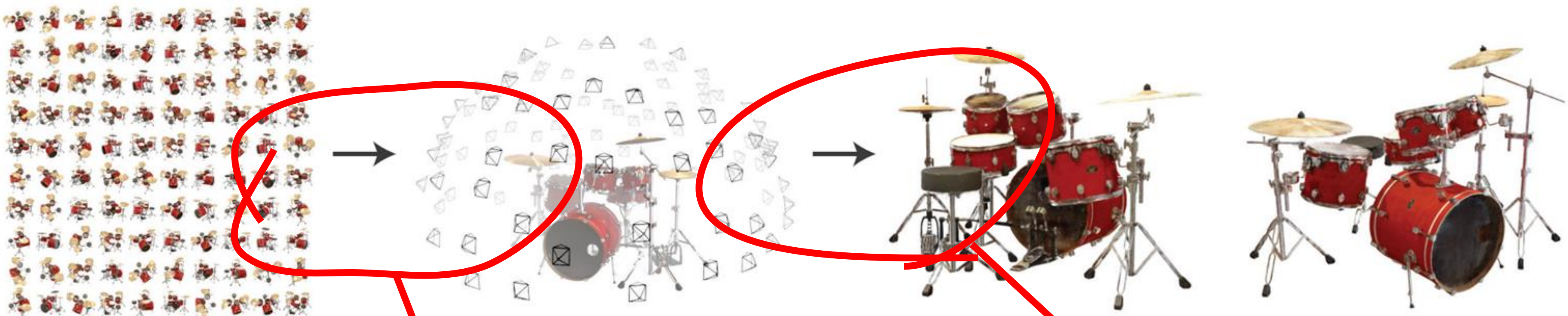


2-D images
(Black Hole Images)

Neural network
create 3-D representations
(Emission)

Images of new views
(reconstruction of
emission at a given time)

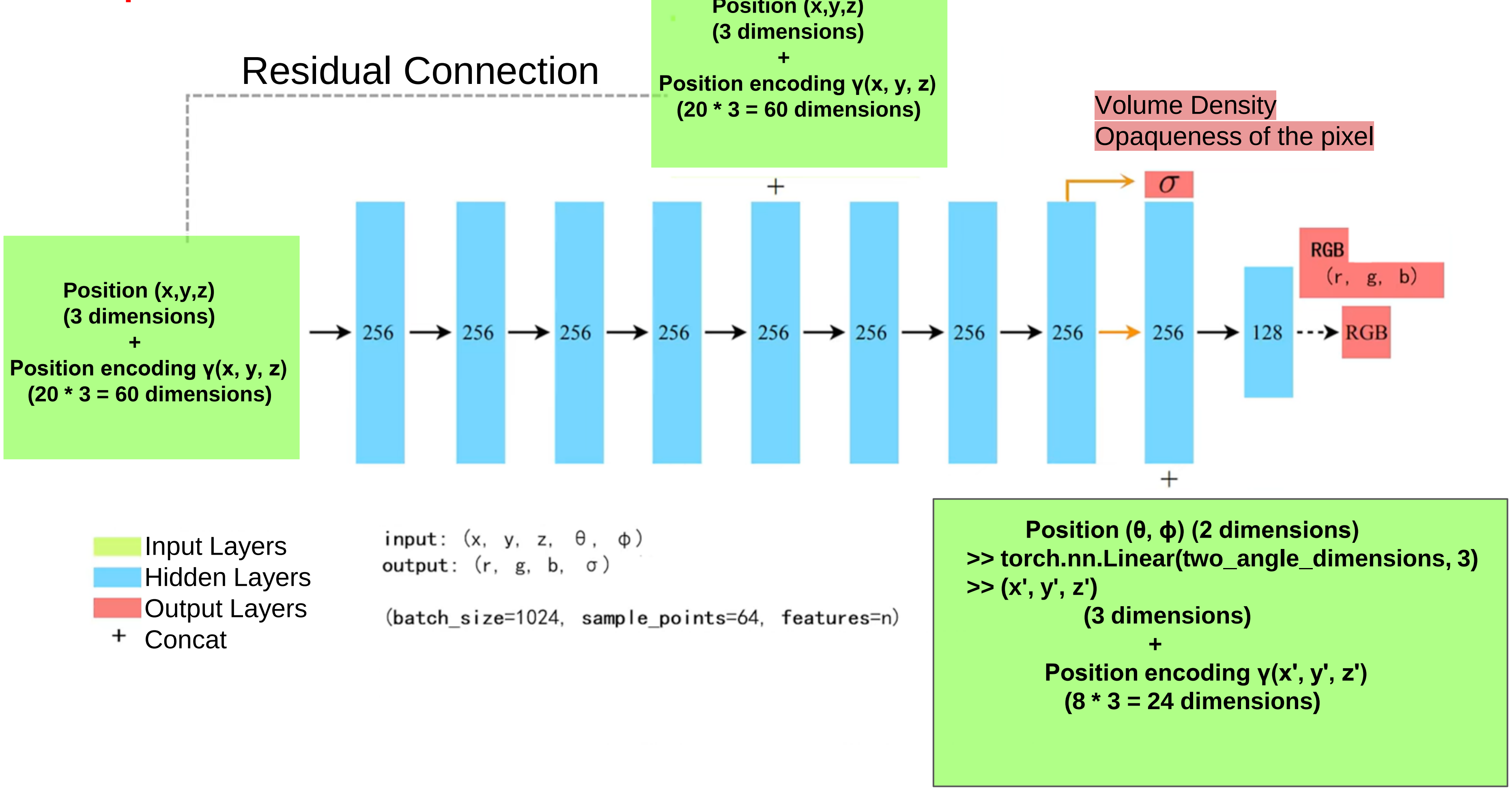
NERF



Step A. Done by a neural network

Step B. Rendering Equation

Step A. Neural Network:



Step B. Rendering Equation

(Discussed in Context)

r(x): R → R^3 – distance on ray to 3D point
σ(**p**): R^3→ R – density at a 3D point
T(t): R → R – transmittance of a location on th eray

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt$$

Full version includes viewing direction
~Correct formula for normal conditions

Step B. Rendering Equation
Volume Rendering

Approximate the integral discretely:

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t),\mathbf{d})dt = \sum\nolimits_{i\in t_n \ to \ t_f} T_i \cdot CDF(\sigma_i)\cdot c_i$$

CDF: a exponential distribution CDF whose parameter is step size of i

Reparameterize the integral by the path x_i rather than time i.

Give $T_i \cdot CDF(\sigma_i)$ a new name $e(x_i)$:

$$= \boxed{\sum\nolimits_{x_i \in Path \ of \ t_n \ to \ t_f} e(x_i)\cdot c_i}$$

End of NERF

A bit Recap:

Step A. Neural network outputting rgb color value c_i and energy loss at each location

Step B. Rendering Equation is a weighted average of rgb color value c_i at each location of a ray path x_i, with the weight being the energy loss e(x_i)

$$= \sum_{x_i \in Path\ of\ t_n\ to\ t_f} e(x_i) \cdot c_i$$

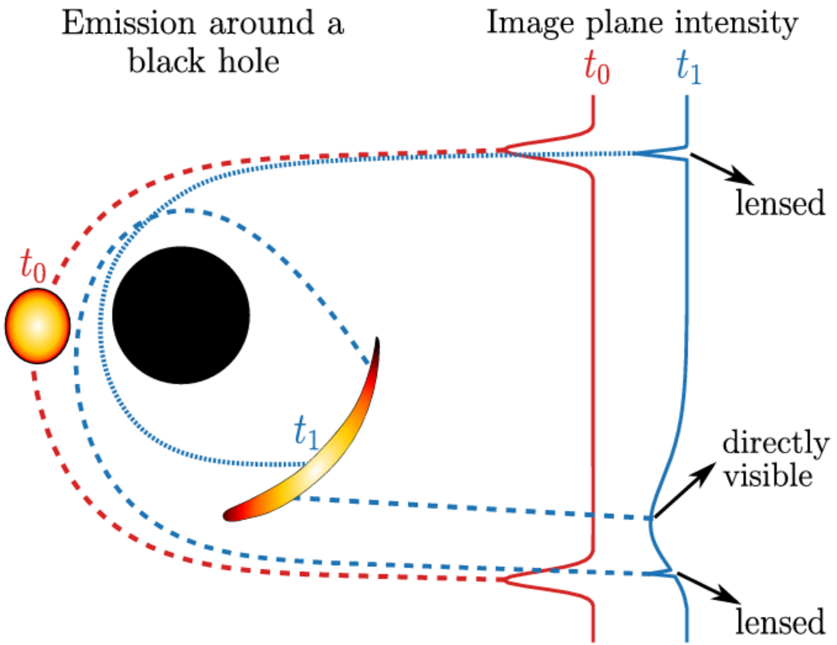
Back to Paper 1:

How is NERF related to Black Hole Tomography?

Input to NERF = Black Hole images (2D)

Output of NERF = Black Hole Emission Tomography (3D)

Is that enough?



1.NERF assumed straight line
v.s.
Curved light around black hole

2.NERF requires multiple viewpoints
v.s.
Single View point of EHT

>> Human knowledge of Keplerian orbital dynamics to overcome the two issues.

Keplerian orbital dynamics

Emission at time t described by emission at time 0:

$$e\left(t, \mathbf{x}\right) = e_0\left(\mathbf{R}_{\boldsymbol{\xi},\phi}\mathbf{x}\right),$$

Φ depends on t

t: time
X: location

R : a rotation matrix that depends on time t and unknown rotation axis ***ξ*** (learned by neural network)

Describing the curved ray path:

$$p_n(t) = \int_{\Gamma_n} e\,(t, \mathbf{x})\,\mathrm{d}s \approx \sum_{\mathbf{x}_i \in \Gamma_n} e\,(t, \mathbf{x}_i)\,\Delta s_i.$$

s: ray
t: time, depends on ray s
X: location, depends on ray s
Γn: the ray path that depends on X, t

Remind you of something?

$$= \sum\nolimits_{x_i \in Path\ of\ t_n\ to\ t_f} e(x_i) \cdot c_i$$

How to learn it with neural network?

$$e_0\left(\mathbf{x}\right) = \mathrm{MLP}_{\boldsymbol{\theta}}(\gamma(\mathbf{x}))$$

$$e\left(t, \mathbf{x}\right) = e_0\left(\mathbf{R}_{\boldsymbol{\xi},\phi}\mathbf{x}\right),$$

$$\longrightarrow \mathrm{MLP}_{\boldsymbol{\theta}}\left[\gamma(\mathbf{R}_{\boldsymbol{\xi}}\mathbf{x}_i)\right]$$

Learn the unknowns with neural networks !

Learn it with neural network:

$$p_n(t) = \int_{\Gamma_n} e\,(t, \mathbf{x})\, \mathrm{d}s \approx \sum_{\mathbf{x}_i \in \Gamma_n} e\,(t, \mathbf{x}_i)\, \Delta s_i .$$

$$\downarrow$$

$$p_n(\boldsymbol{\theta}, \boldsymbol{\xi}) = \sum_{\mathbf{x}_i \in \Gamma_n} \mathrm{MLP}_{\boldsymbol{\theta}} \left[\gamma(\mathbf{R}_{\boldsymbol{\xi}} \mathbf{x}_i)\right] \Delta s_i .$$

What are labels for the model?

Complex Visibilities (from telescope):

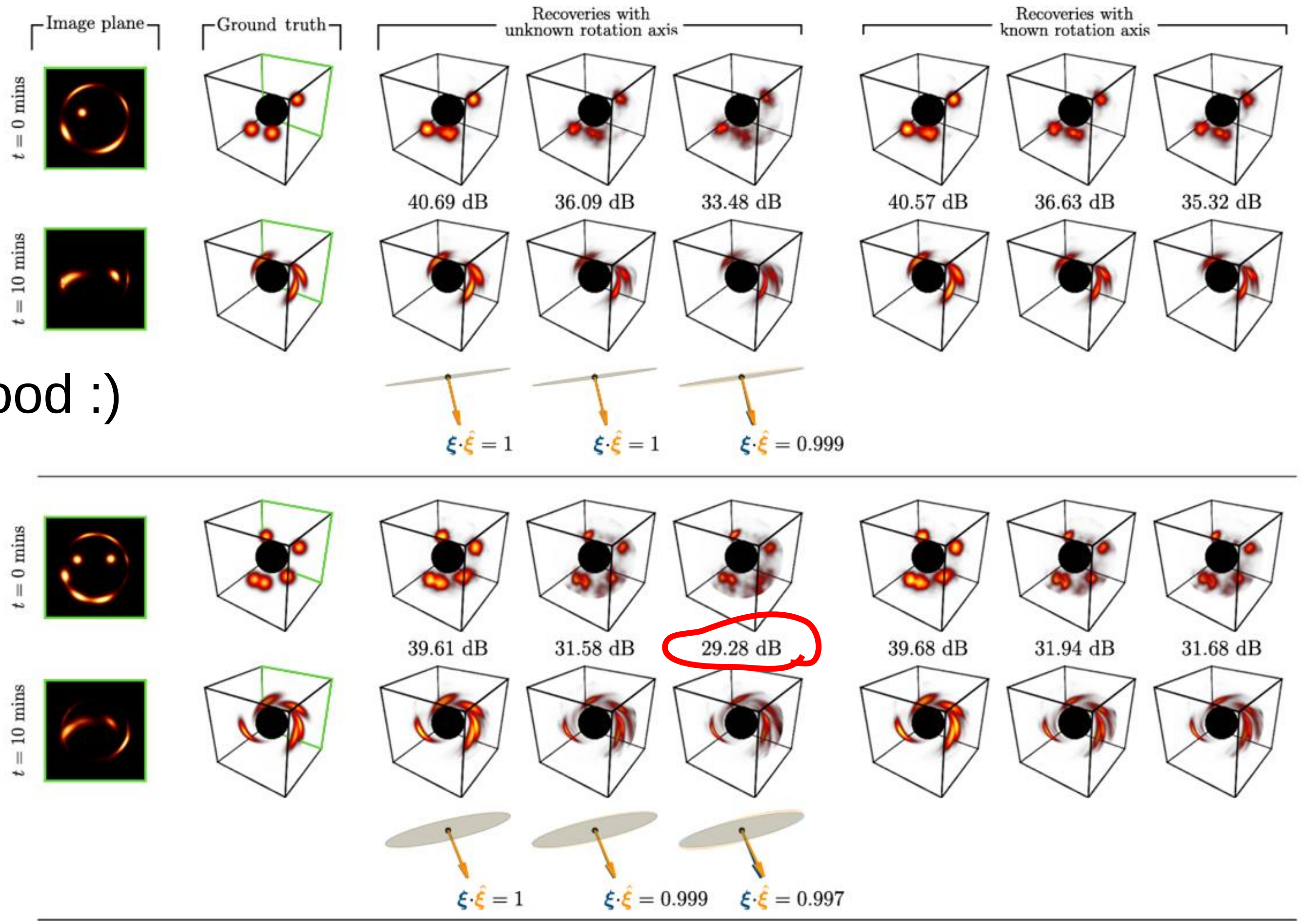
$$\mathbf{y}(t) = \mathbf{F}_t \mathbf{I}(t) + \varepsilon.$$

I(t): Image, matrix of P_n
F: Fourier transformation with additional constraints from telescopes
ε: Noise

The loss function is squared error across time t
(the norm also incorporates telescope data)

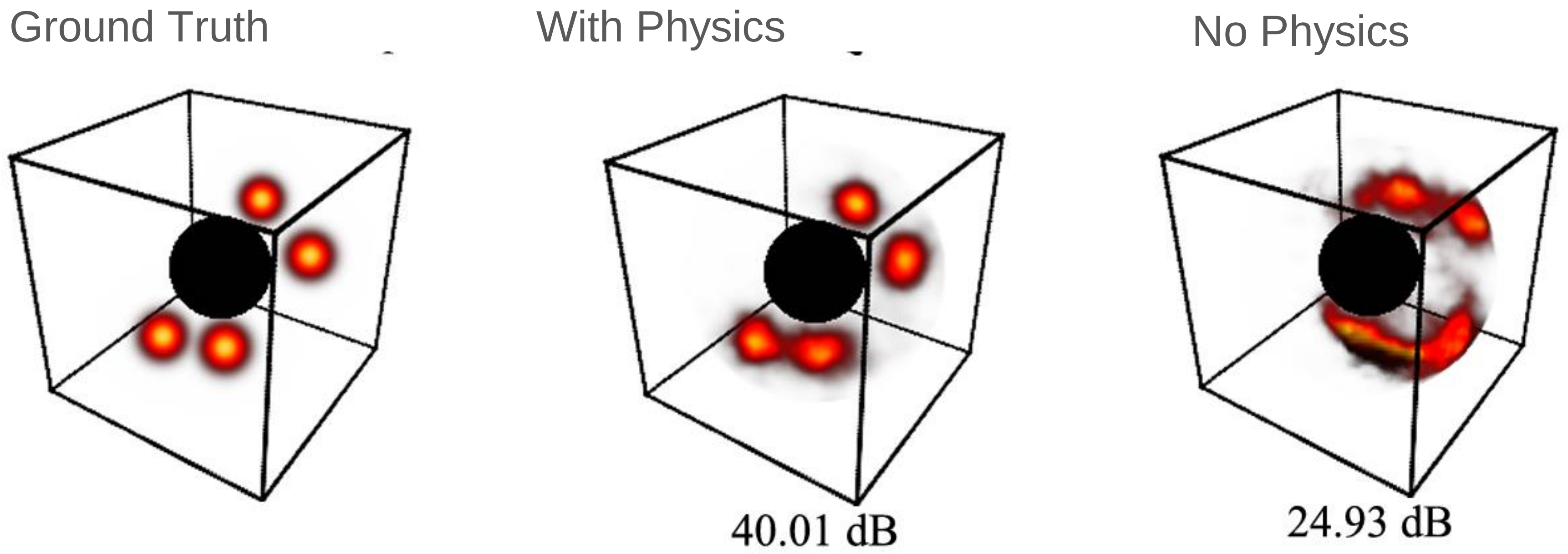
Result: Compare reconstruction with ground truth

In Quantitative PSNR:
30 dB and above = Good :)



A Good Lesson:

Apply NERF to black hole emission reconstruction:
Use physics to compensate insufficient data
CV is not about put everything into neural networks!



Method Critique of Paper [1]

Overall: Very different from standard CV paper. Plenty of math…

>>Generalize the idea of NERF to a greater setting

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt = \sum_{i \in t_n \ to \ t_f} T_i \cdot CDF(\sigma_i) \cdot c_i$$

CDF: a exponential distribution CDF whose parameter is step size of i

Reparameterize the integral by the path x_i rather than time i.

Give $T_i \cdot CDF(\sigma_i)$ a new name $e(x_i)$:

$$= \boxed{\sum_{x_i \in Path \ of \ t_n \ to \ t_f} e(x_i) \cdot c_i}$$

Use domain knowledge (orbital dynamics) to overcome difficulties

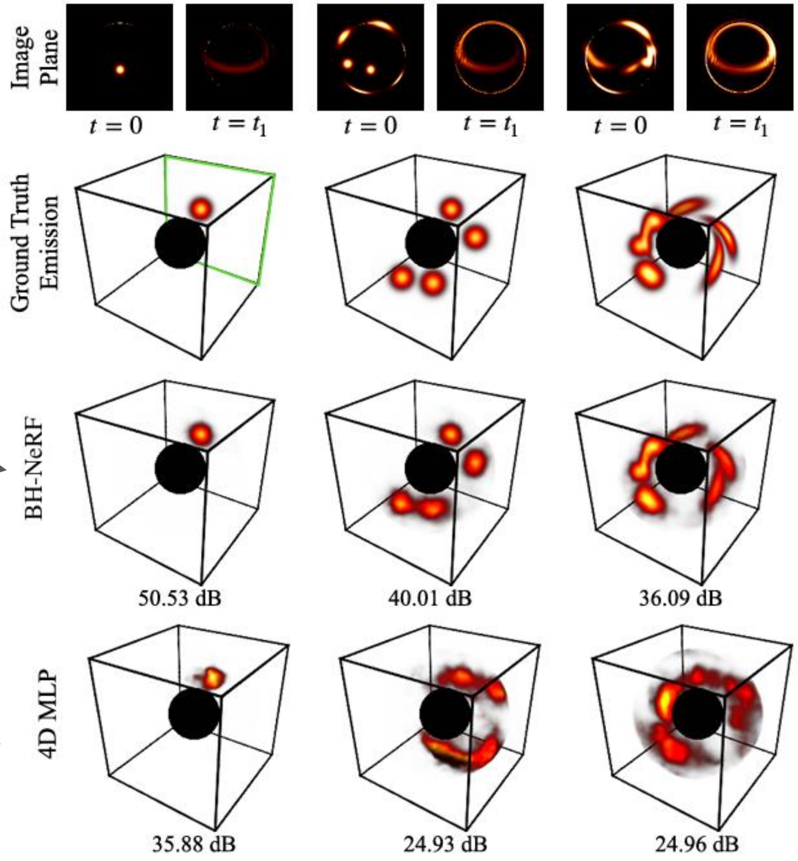
$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\xi}) = \sum_t \|\mathbf{y}(t) - \mathbf{F}_t \mathbf{I}_{\boldsymbol{\theta},\boldsymbol{\xi}}(t)\|^2_{\boldsymbol{\Sigma}} ,$$

Additional data from telescope
(but not image data)

Good demonstration on
the necessity of orbital dynamics
in the model

With orbital dynamics

Without orbital dynamics



Future Works

[1] used many assumptions about the model, might not work very well under different conditions (instrumental errors and atmospheric errors)

[2] decided Manhattan Distance for Similarity Search.
Can we instead learn a distance function? >> Metric learning

Matching Networks, Protonets

Fine-tuning/transfer learning >> Few-shot learning

Source: [2] Walmsley, Mike, et al. "Practical galaxy morphology tools from deep supervised representation learning." Monthly Notices of the Royal Astronomical Society 513.2 (2022): 1581-1599.