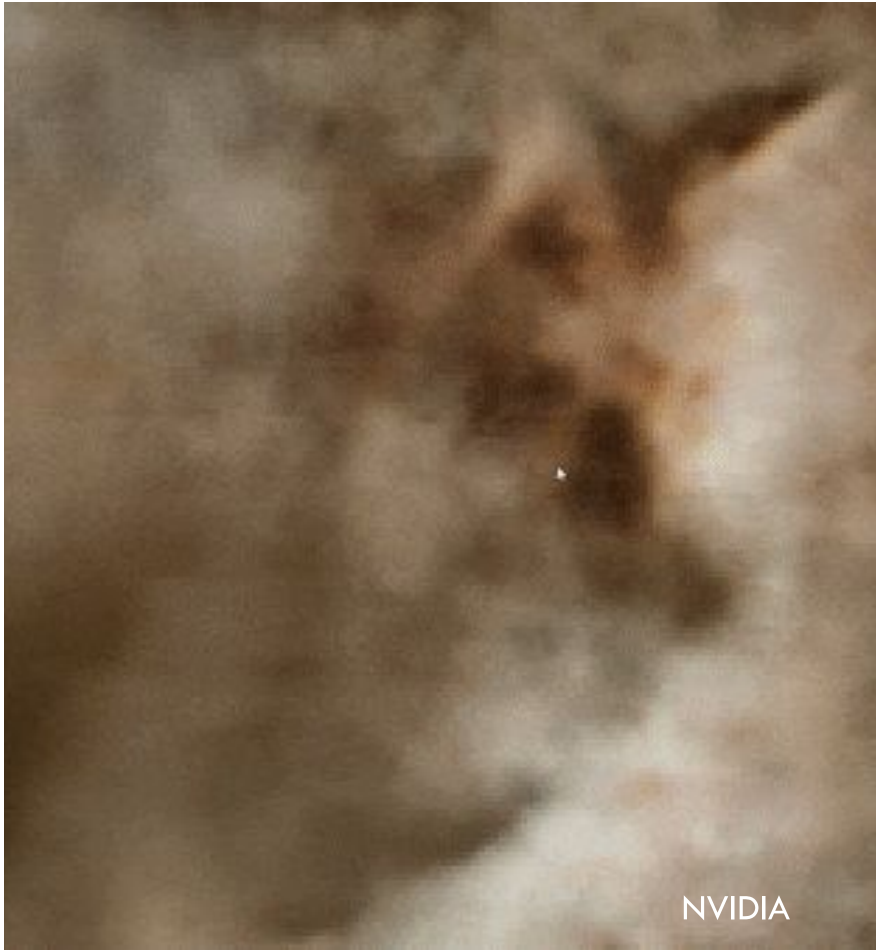
Learning the Real World

An Introduction to Neural Implicit Representations

Jack Naylor

MTRX8700 Student Lectures
2pm 28/4/22

THE UNIVERSITY OF SYDNEY



Neural Implicit Representations

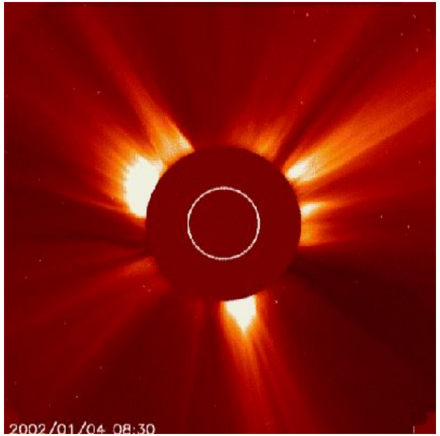
THE UNIVERSITY OF SYDNEY

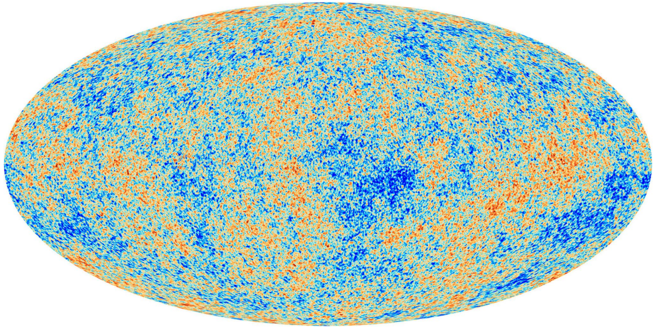
Nature

Luckily for us: we live in a continuous world.

Things are smooth, differentiable and explainable by physics!

Nature is smooth and continuous!



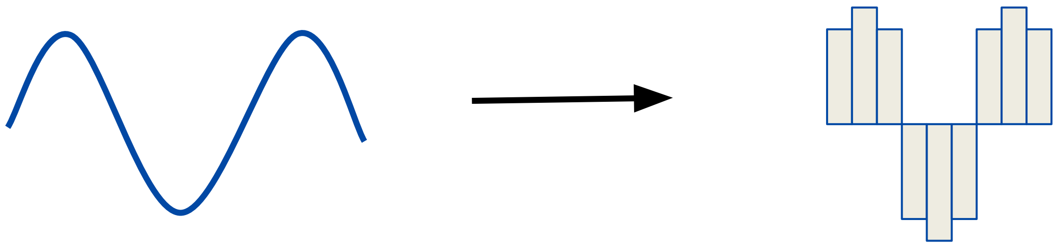


Sound

Light + Fluids

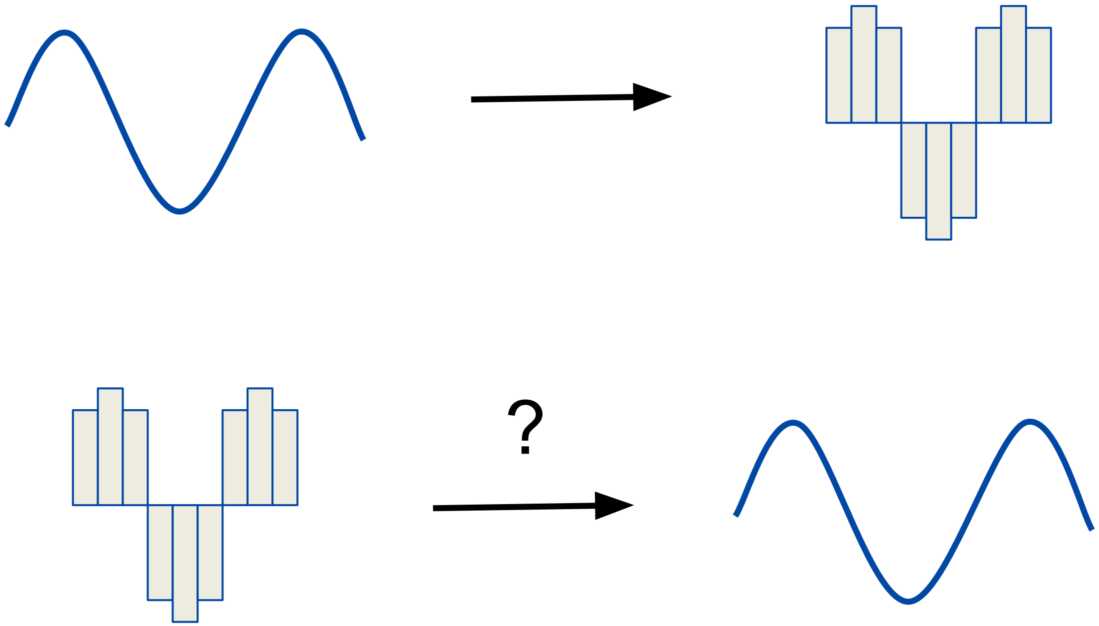
Even the CMBR

Implicit Neural Representations

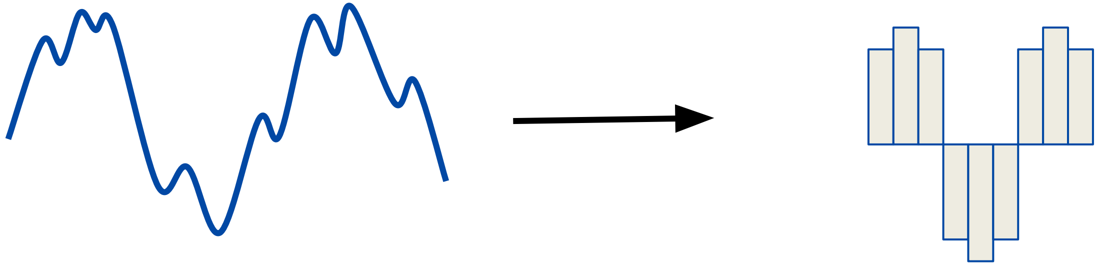


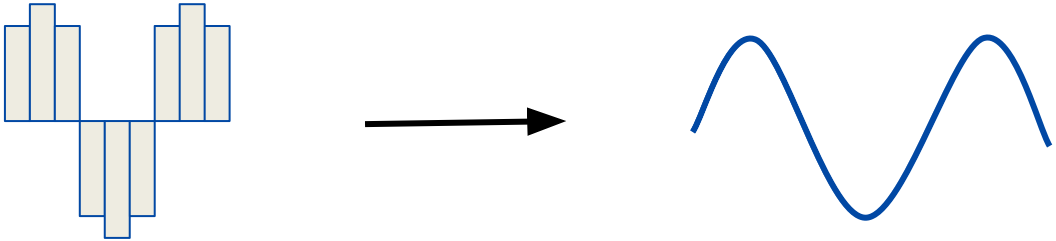
Measure Continuous Functions as Discrete Samples

Implicit Neural Representations



Cannot Always Reconstruct Difficult Continuous Functions from Discrete

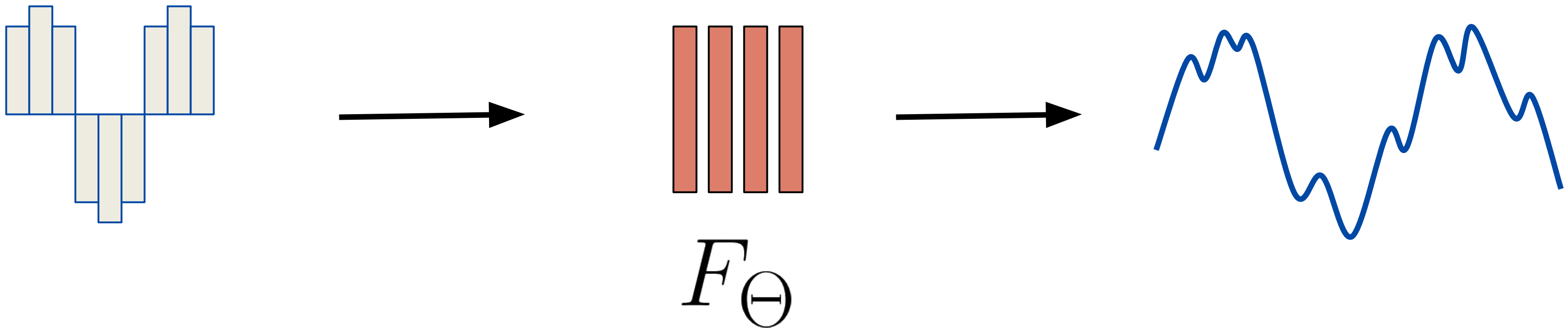




Not Unique if Undersampled!

No!

Implicit Neural Representations

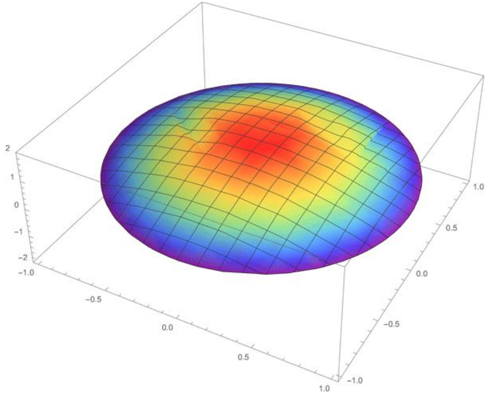


Learn an implicit function to approximate the continuous signal

Common Discretised Signals

Pixels are a discrete space

Video has pixels and a framerate (temporally discrete)



Meshes, pointclouds and PDE's all have discrete domains

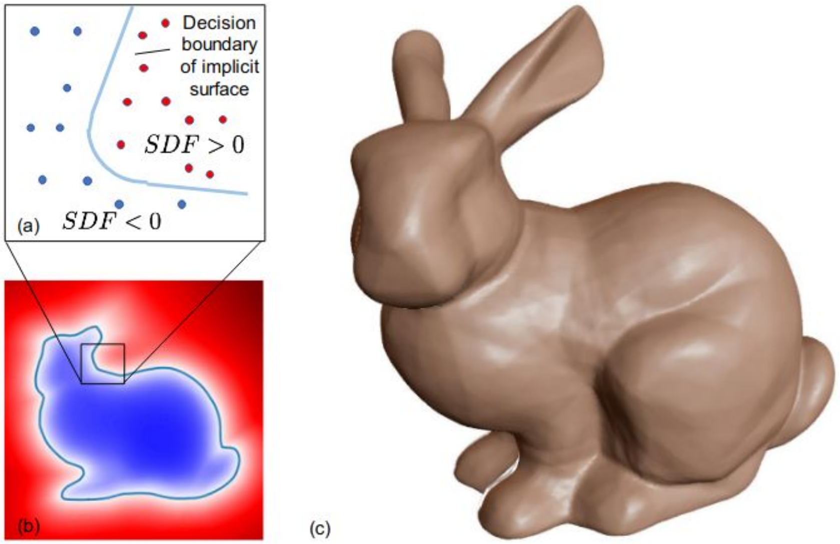
NFSA, NASA and Purdue University

An example: DeepSDF

- The simplest case: learn where a surface is.

- Discretise 3D space, sample points and say whether inside, or outside the bunny.

- Learn a continuous, smooth surface which separates physical regions.



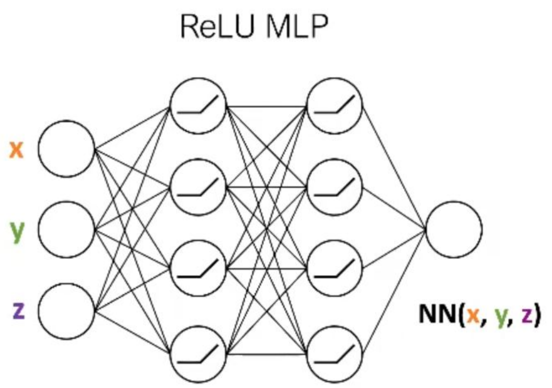
Park et al. 2019

What sort of network do we need?

$$F(\mathbf{x}, \Phi, \nabla_{\mathbf{x}}\Phi, \nabla^2_{\mathbf{x}}\Phi, \ldots) = 0, \; \Phi : \mathbf{x} \mapsto \Phi(\mathbf{x})$$

Approximate *some* function

An MLP works as a function approximator, and by Cybenko's theorem: there exists an MLP of sufficient dimension which can approximate our function well enough.



Using *some* nonlinear activation function

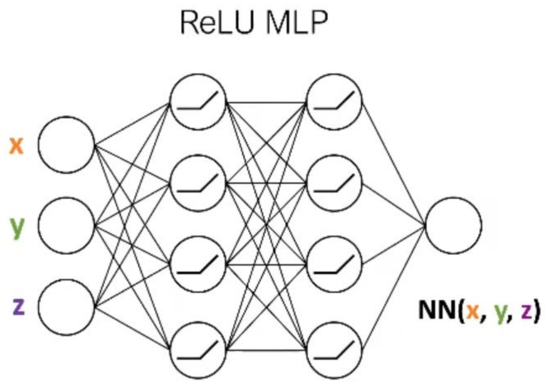
ReLU? Step? Leaky ReLU?

What sort of network do we need?

$$F(\mathbf{x}, \Phi, \nabla_{\mathbf{x}}\Phi, \nabla^2_{\mathbf{x}}\Phi, \ldots) = 0,\ \Phi : \mathbf{x} \mapsto \Phi(\mathbf{x})$$

Approximate *some* function

An MLP works as a function approximator, and by Cybenko's theorem: there exists an MLP of sufficient dimension which can approximate our function well enough.



Using *some* nonlinear activation function

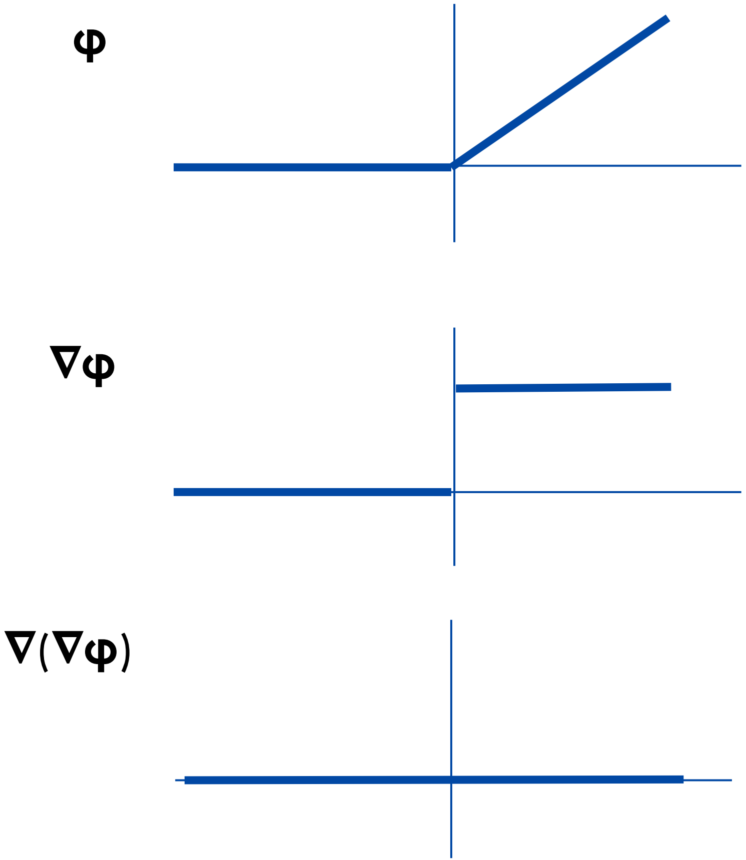
ReLU? Step? Leaky ReLU?

What about sin?

φ and ∇φ and ∇(∇φ) and …

Differentiability of continuous functions is key!

A ReLU's 2nd derivative is 0 - similar for many other nonlinear activation functions!

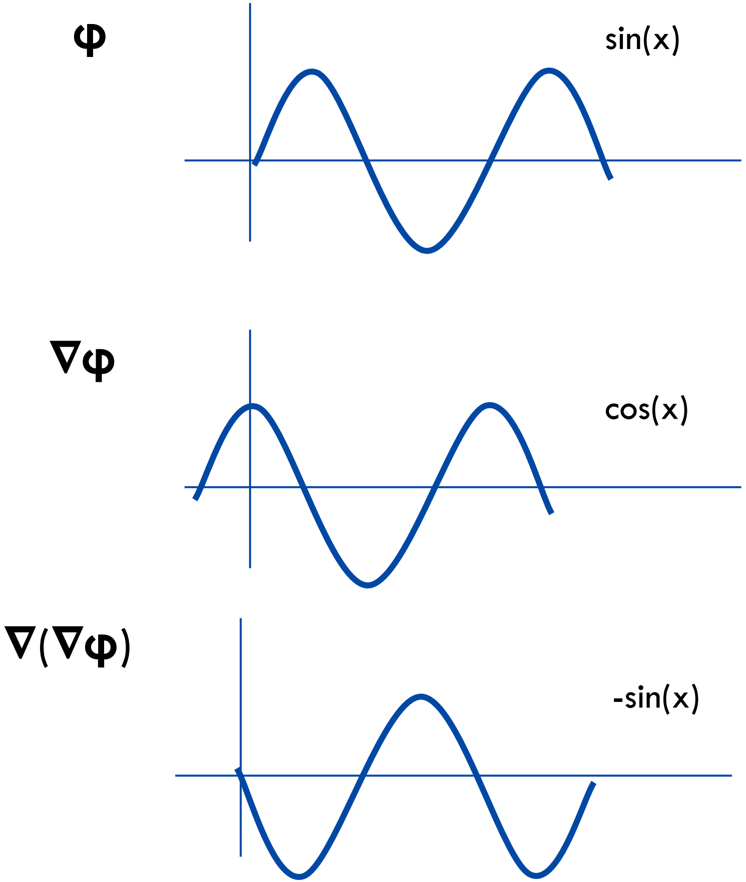


φ and ∇φ and ∇(∇φ) and …

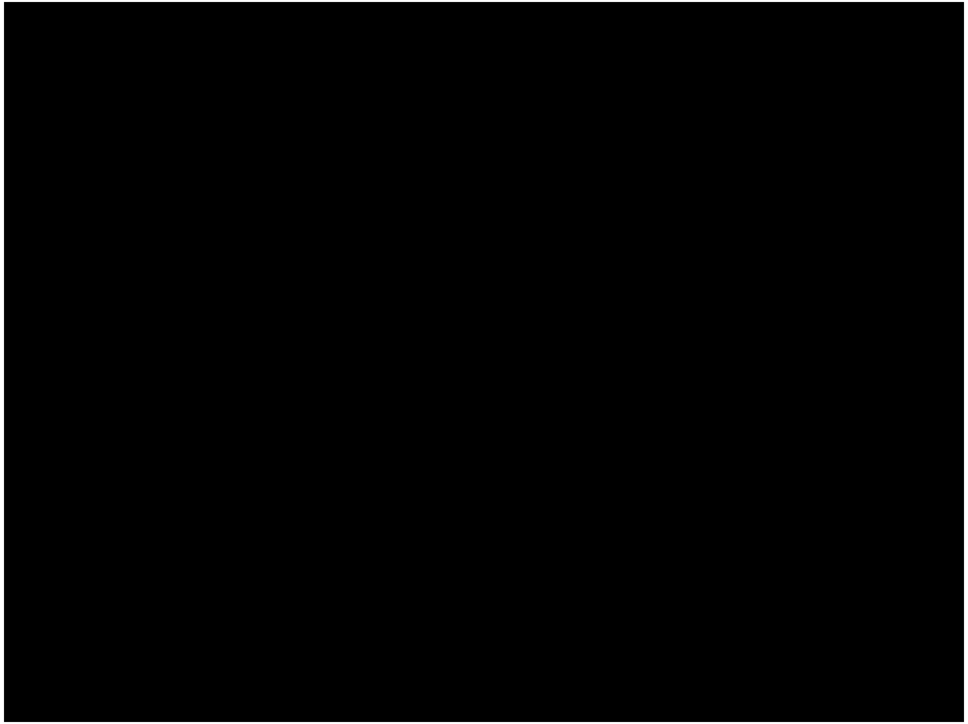
Differentiability of continuous functions is key!

Sine functions are continuously differentiable!

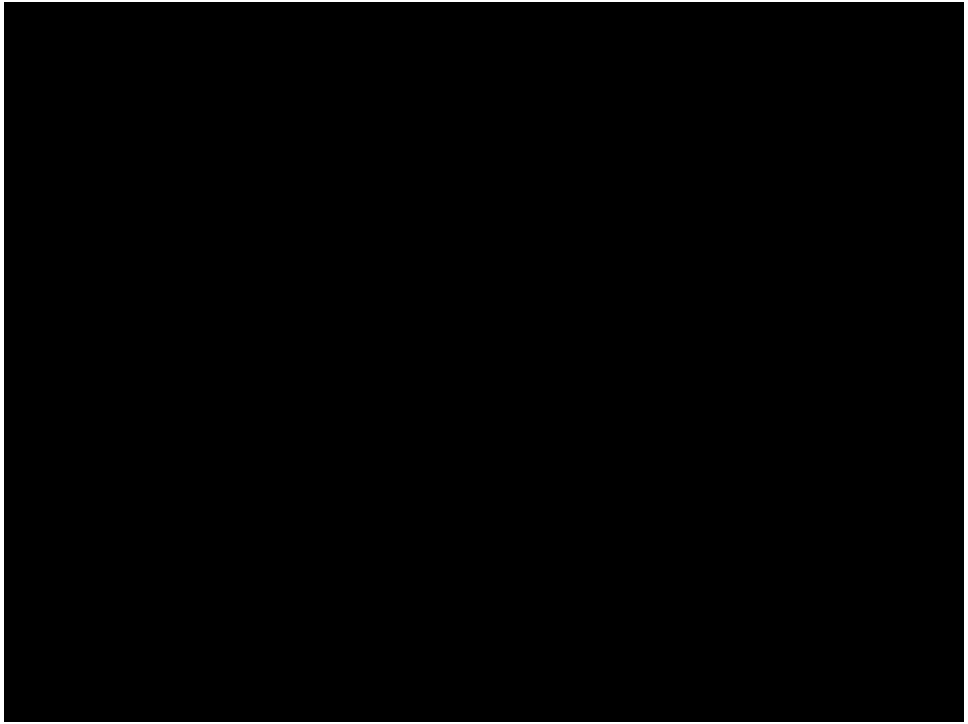
We can model information of higher orders! Higher frequencies!



SIREN



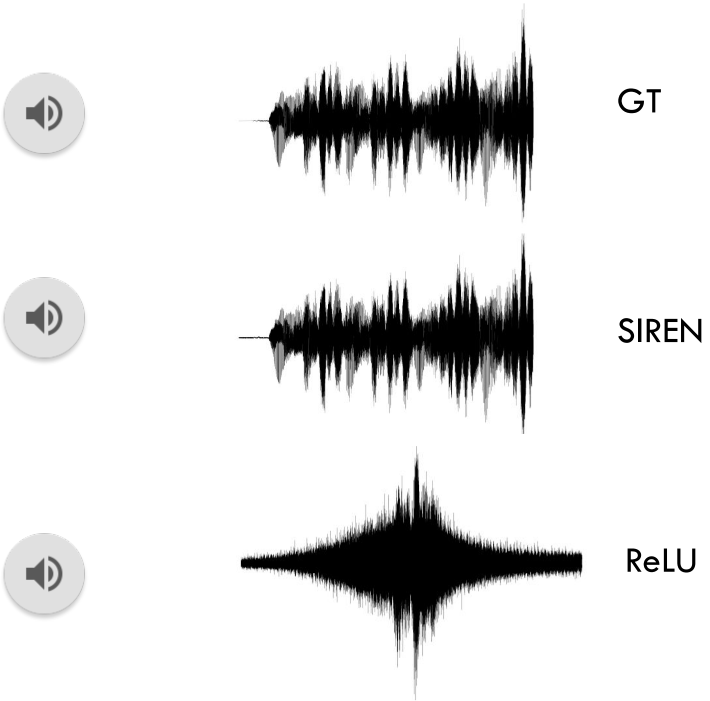
Sitzmann et al. 2020



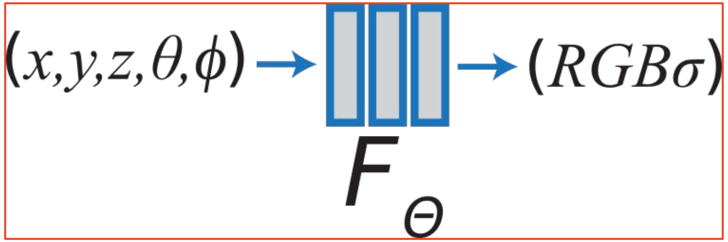
Sitzmann et al. 2020

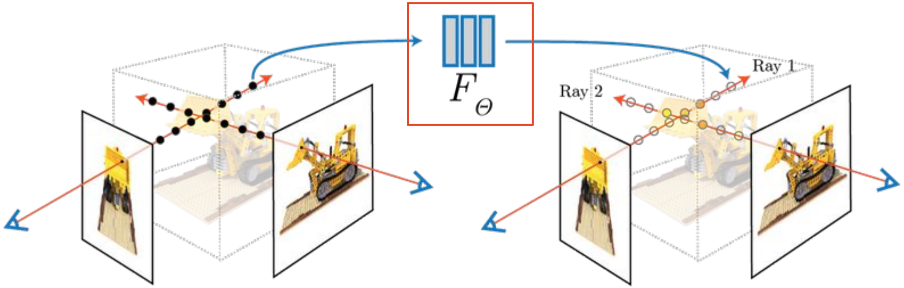
Why does SIREN work?

- **Underlying smoothness to derivatives**
- **Derivative of a SIREN is a SIREN i.e. decision making with derivatives.**
- **Pseudo-Fourier decomposition**



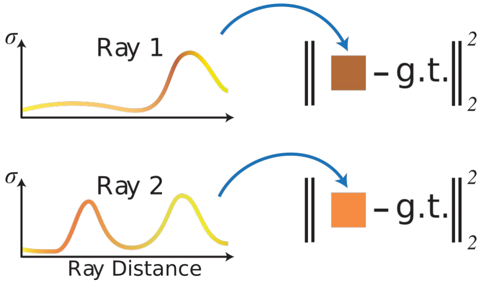
NeRF: Modelling Light





Mildenhall et. al (2020)

- Light is continuous!
- Use a network to learn a continuous, volumetric 5D light field!
- Why does NeRF not use SIREN?

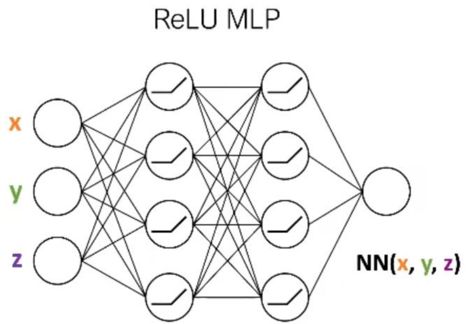


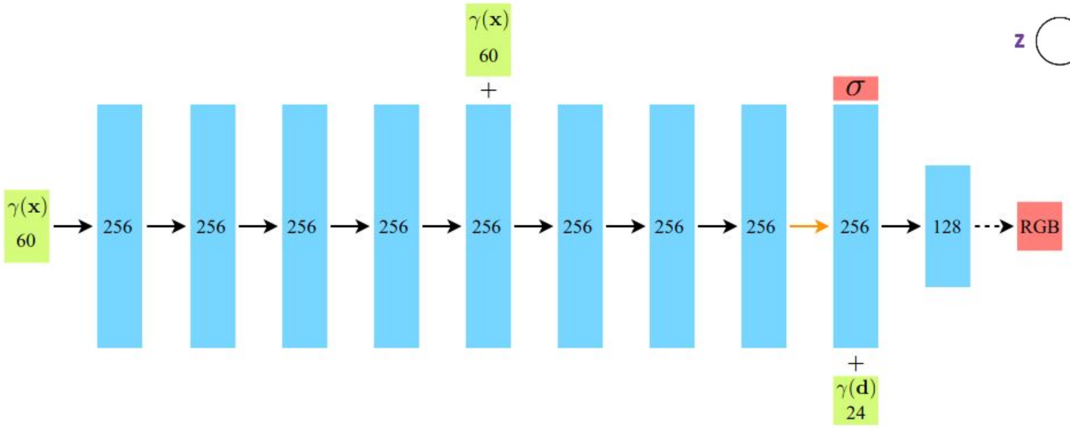
Different light fields!

Depth for free!
(We learn a volume!)

Positional Encoding

The Key to NeRF

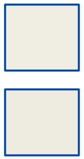


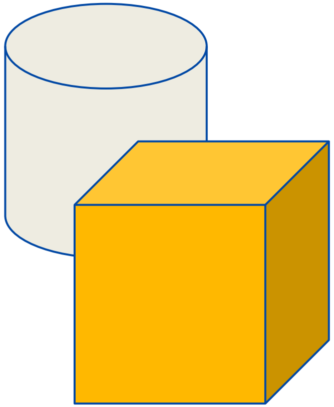


$$\gamma(\mathbf{x}) = \left(\sin(2^0\pi\mathbf{x}), \cos(2^0\pi\mathbf{x}), \ldots, \sin(2^{L-1}\pi\mathbf{x}), \cos(2^{L-1}\pi\mathbf{x})\right)$$

Mildenhall et al. 2020

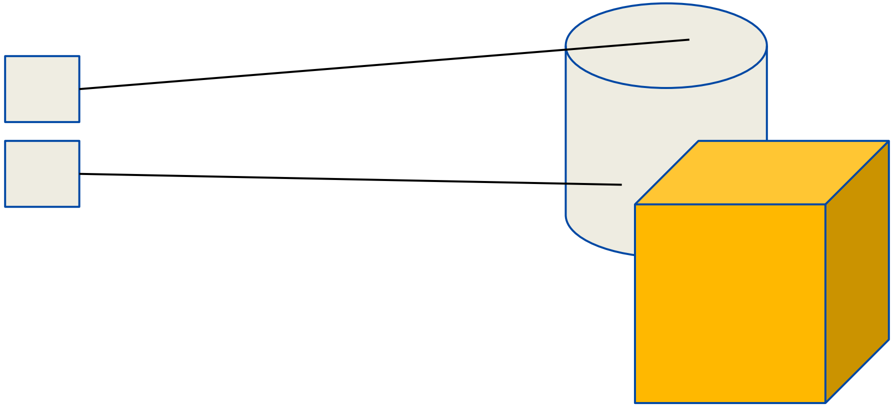
Compare the pair?





Compare the pair?

Virtually the same to us!

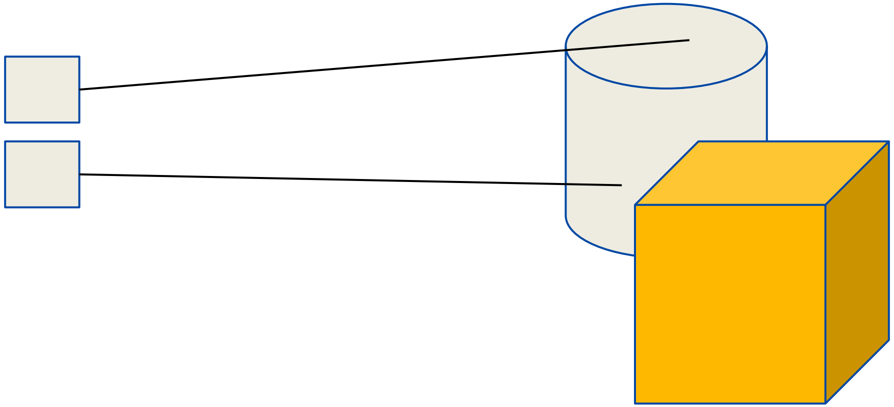


Compare the pair?

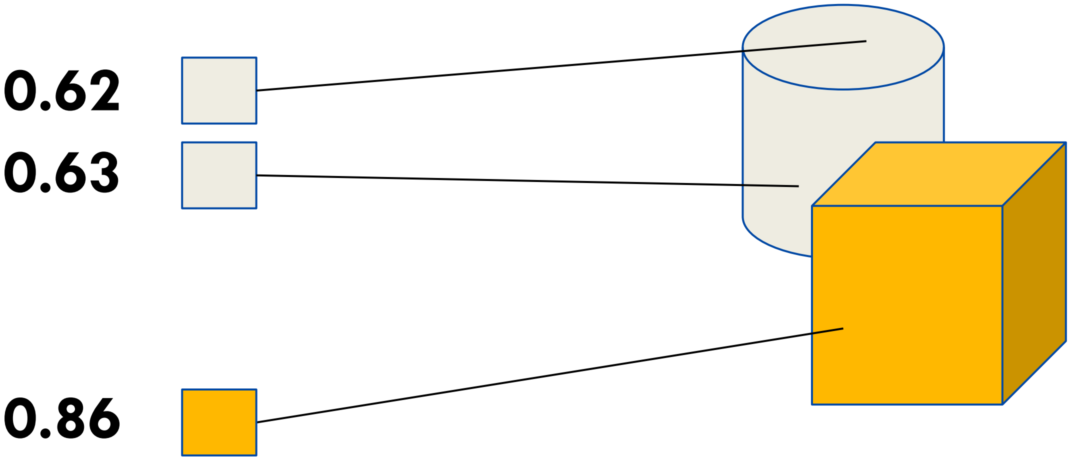
Virtually the same to a computer!

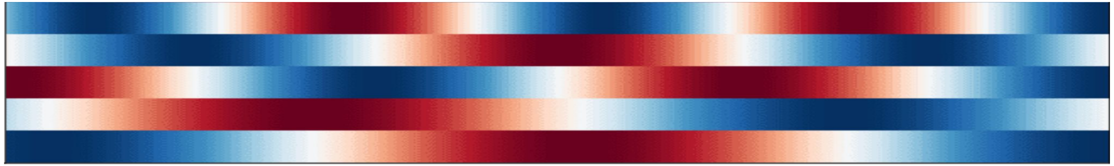
0.62

0.63

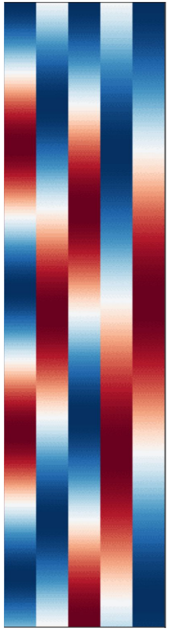


Compare the pair?

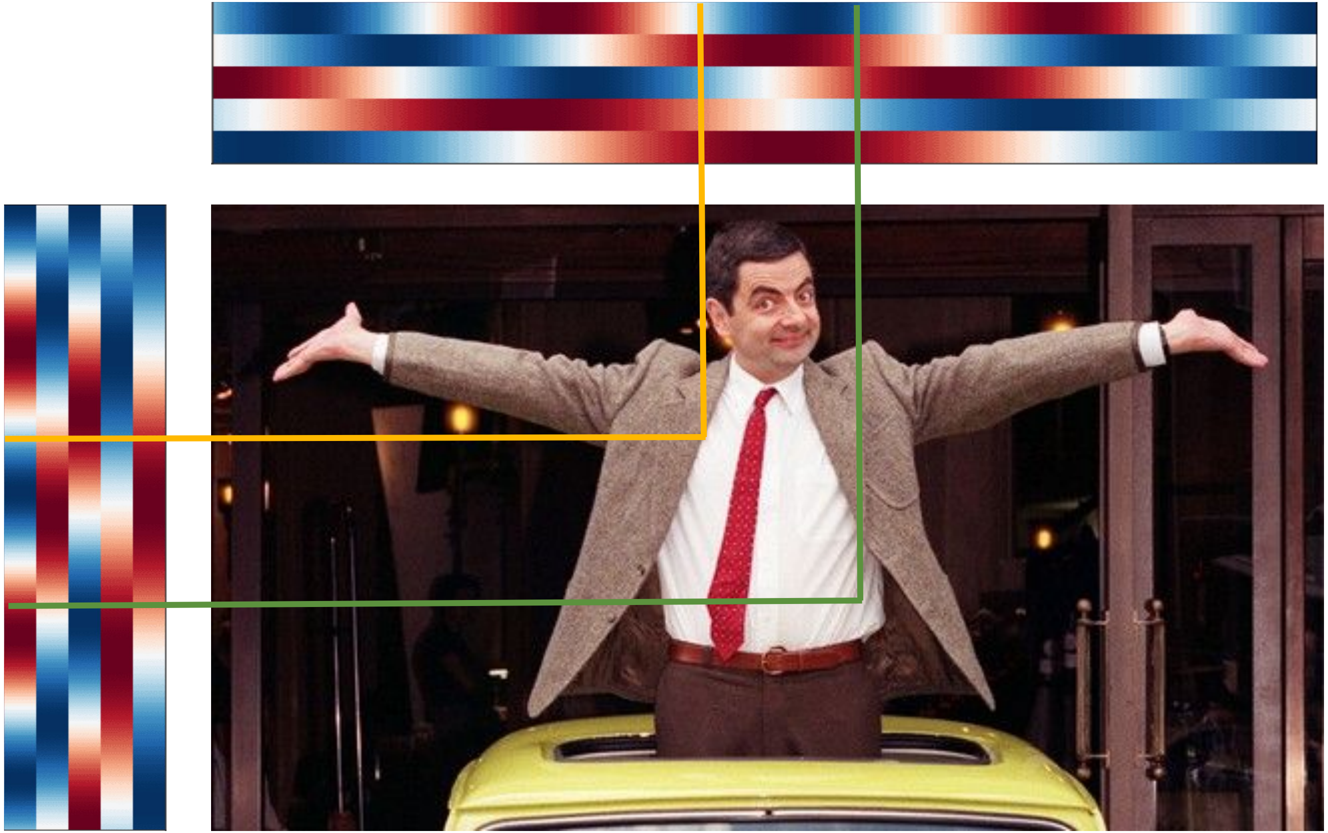




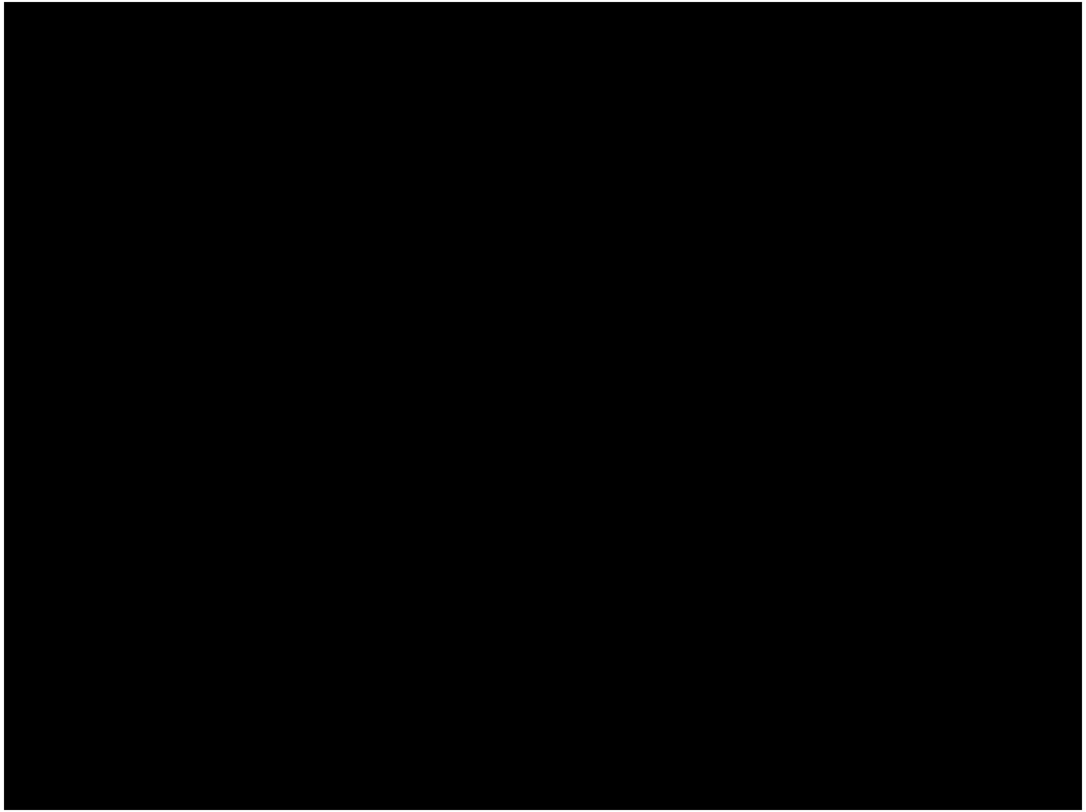
Sinusoids of different frequency



$$\gamma(\mathbf{x}) = \left(\sin(2^0\pi\mathbf{x}), \cos(2^0\pi\mathbf{x}), \ldots, \sin(2^{L-1}\pi\mathbf{x}), \cos(2^{L-1}\pi\mathbf{x})\right)$$



The result? Fourier Positional Encoding!

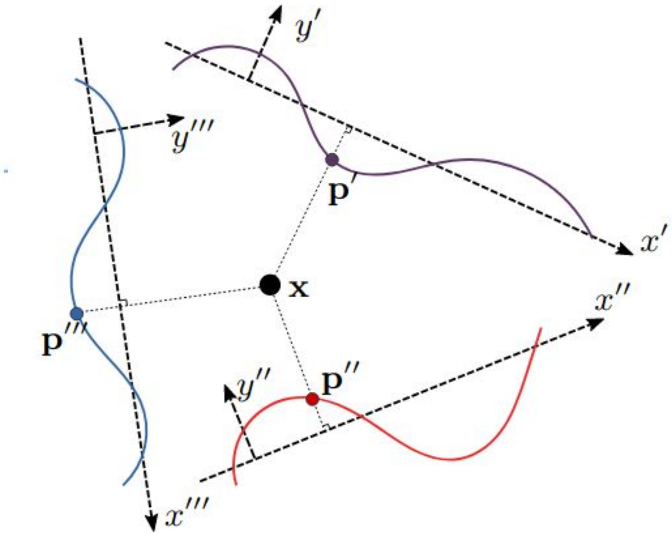


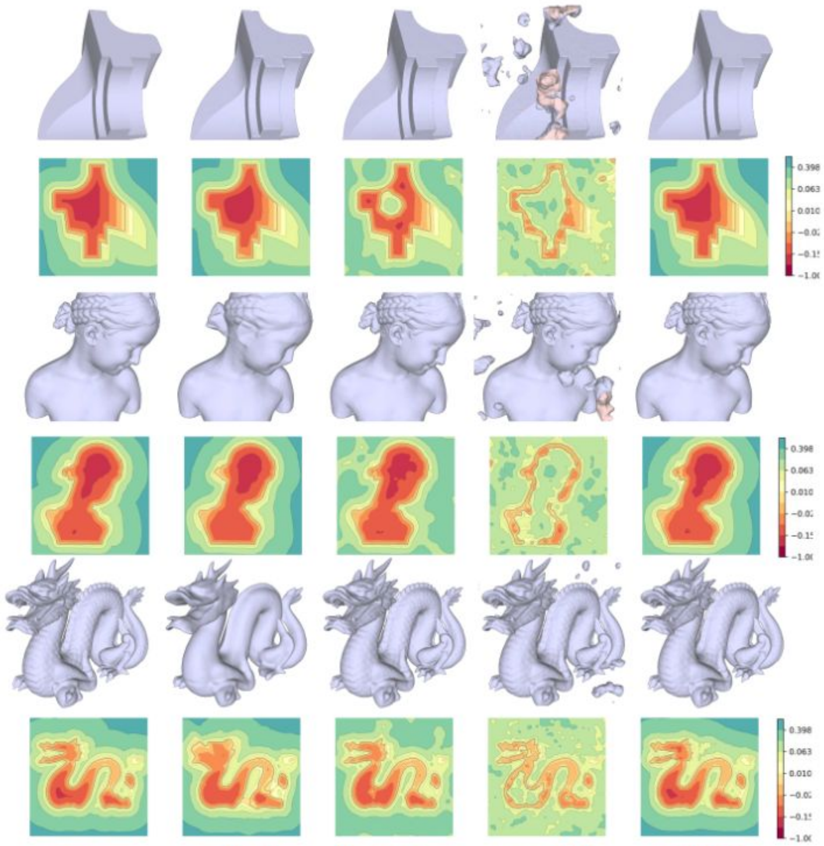
Some obvious questions

- Is this just a NeRF thing?
- No! Uniquely encoding positions is now widely used in neural implicit functions

- Does it need to be sinusoids?
- No! In fact, gaussians and spherical harmonics work better.

SDFs w/ Spline Encoding



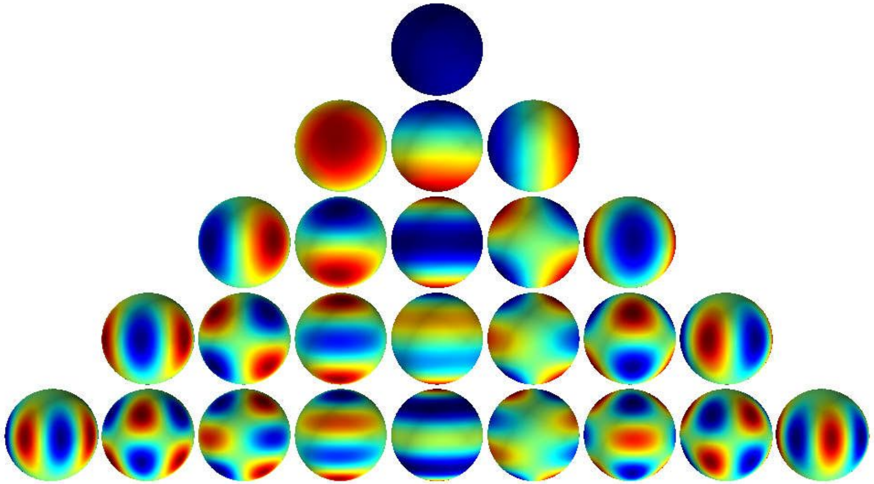


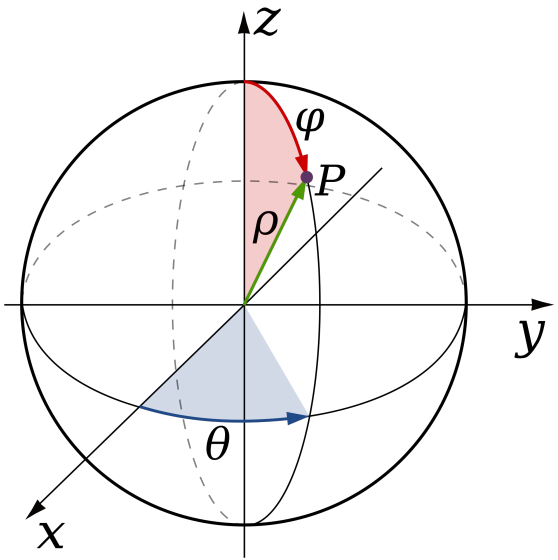
Wang et al. 2019

Some obvious questions

- Is this just a NeRF thing?
- No! Uniquely encoding positions is now widely used in neural implicit functions

- Does it need to be sinusoids?
- No! In fact, gaussians and spherical harmonics work better.





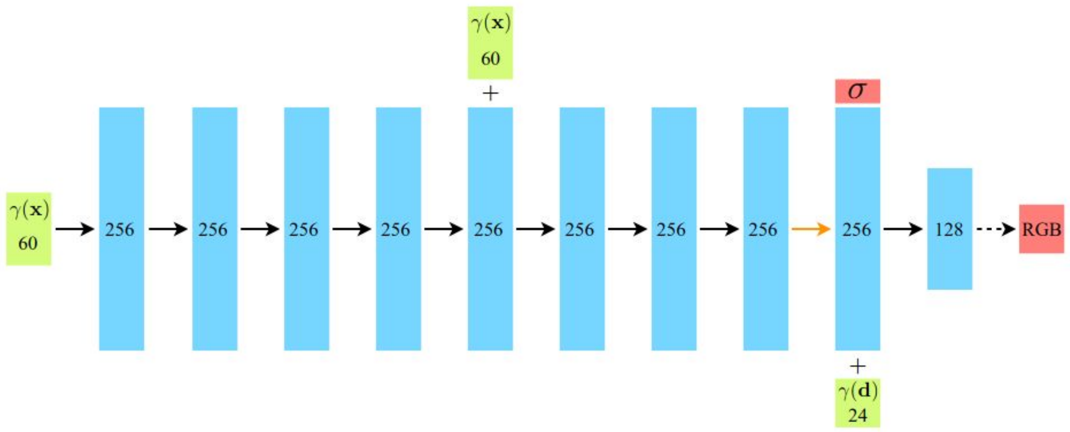
- Parameterise theta, phi over the sphere
- 2D lookup (u,v) to (theta,phi)

Modelling the Real World

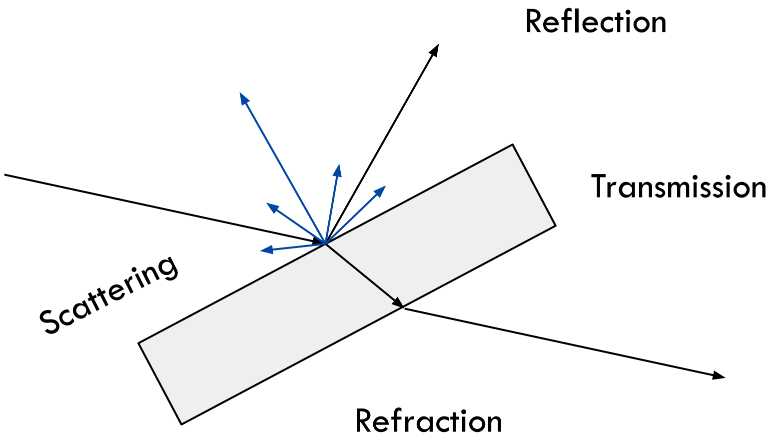
THE UNIVERSITY OF SYDNEY

Physics

- Neural networks know nothing about the real world!
- Fortunately, we've got a few hundred years of understanding physics and nice maths!
- What is NeRF modelling?

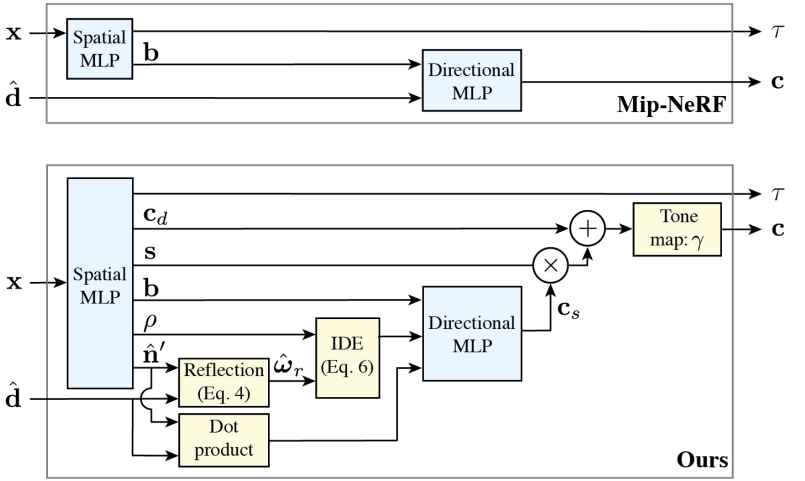


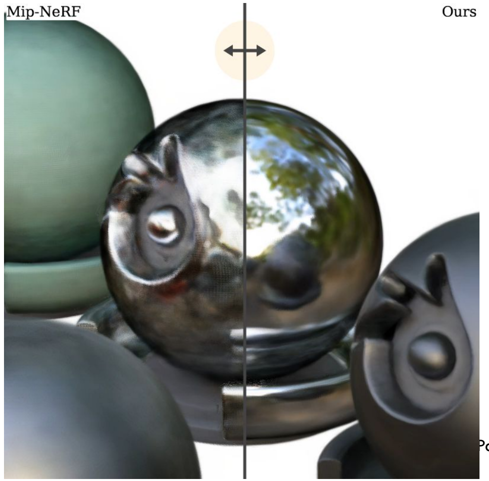
How Light Usually Works



Learning under physics

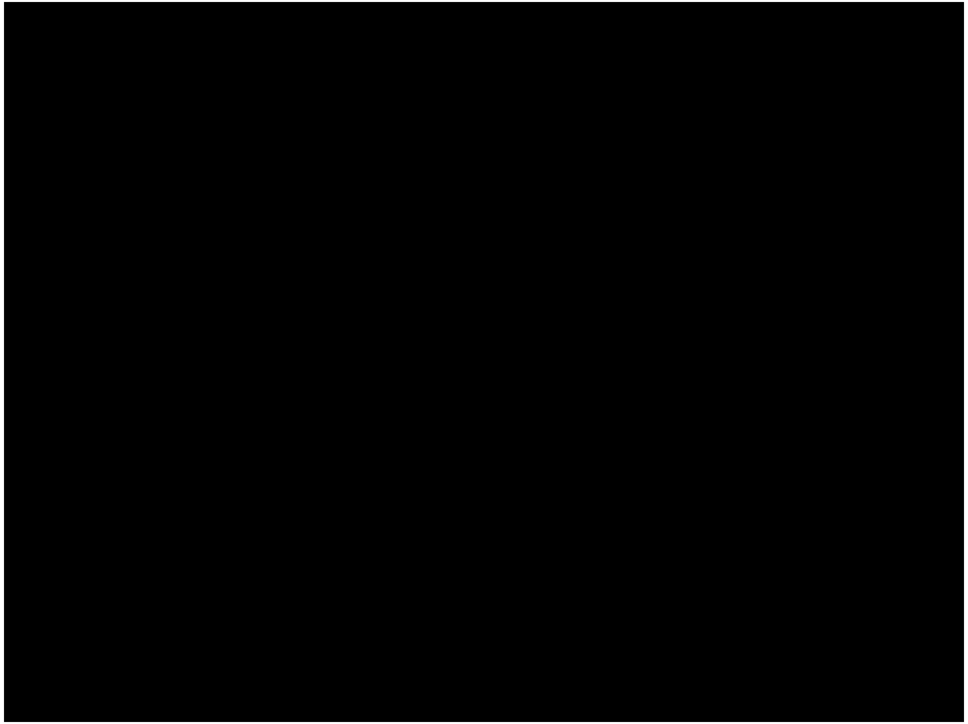
- Give the network some physics!
- Reflections change ray direction





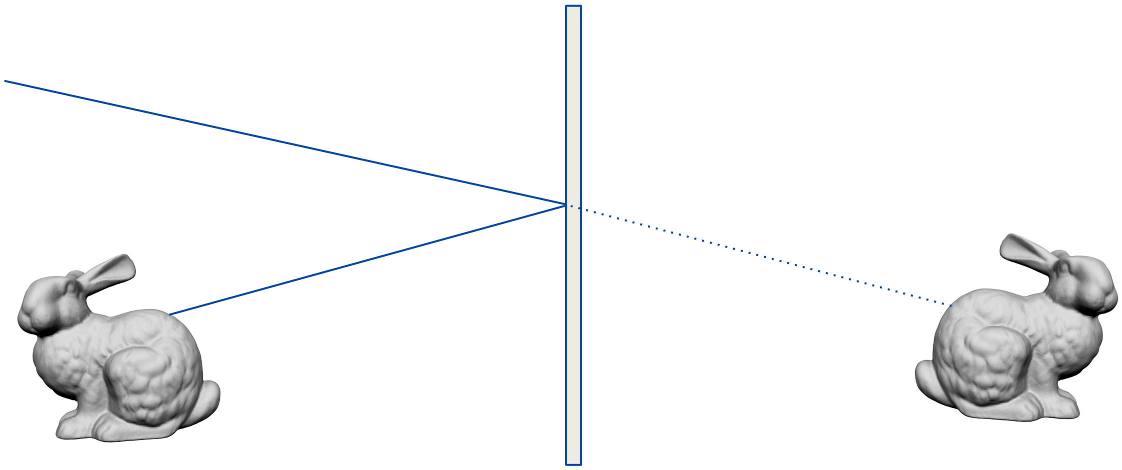
Verbin et al. 2022

House of Mirrors

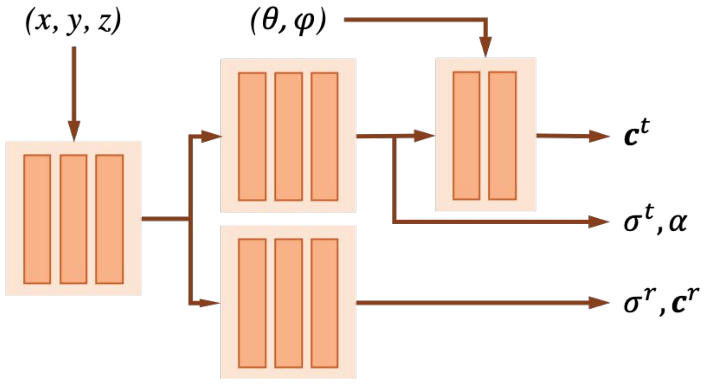


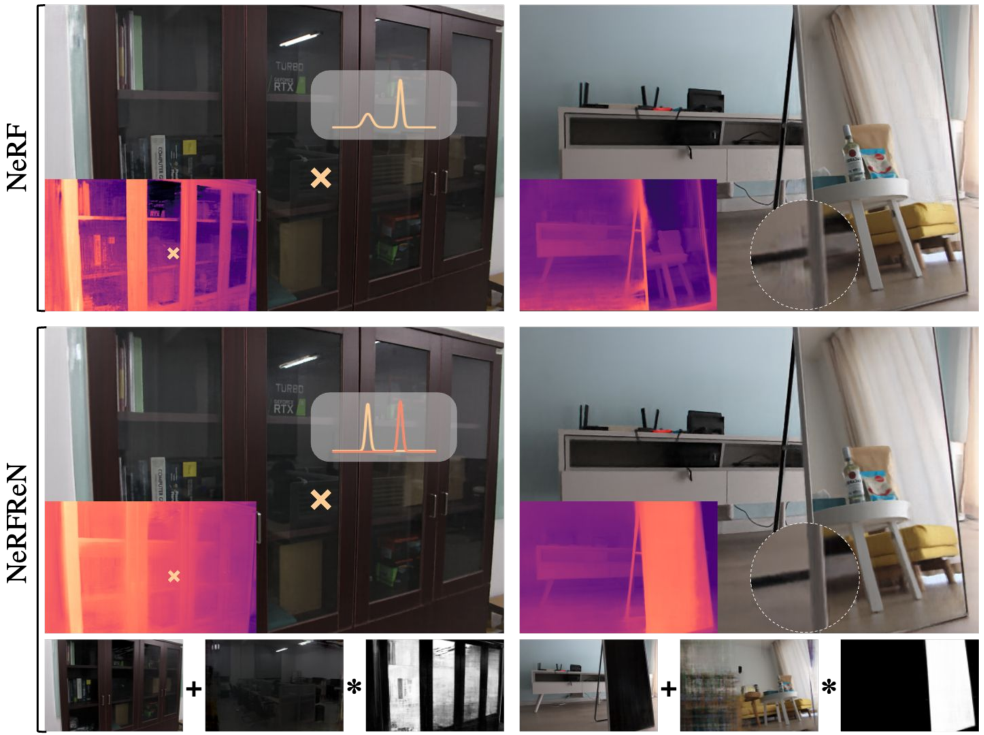
Mirrored Worlds

A NeRF is making sense of it, the best way it knows!



Model reflections!



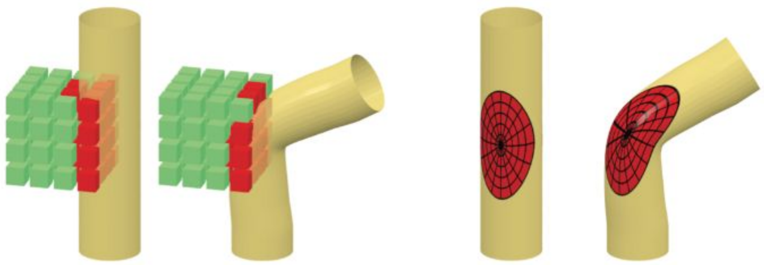


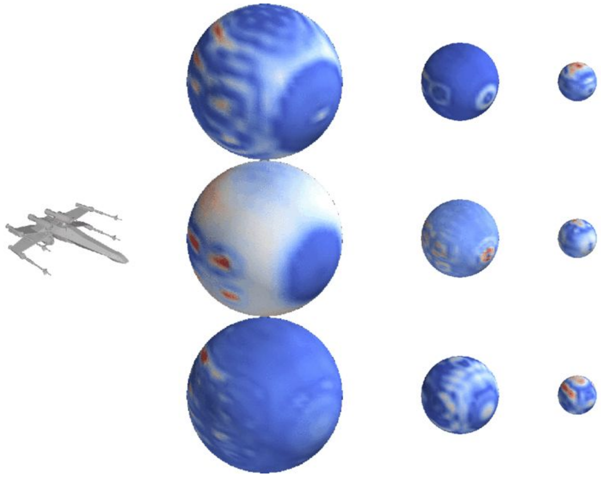
Guo et al. 2021

Why no convolutions?

- Given the network an invariant framework (e.g rays have no relation).
- Convolutions and graphs work on discrete frameworks - we want a continuous function.
- Sounds like something to do with geometry…

Masci et al. 2016





Cohen et al. 2018

Neural Implicit Representations

↑ Can we process?

Geometric Deep Learning

Neural Implicit Representations

Can we process?

Geometric Deep Learning

Another lecture for
another day…

Q&A

Jack Naylor

jack.naylor@sydney.edu.au

THE UNIVERSITY OF SYDNEY