Everything you wanted to know about NeRFs.

Reading Group, April 7th

Filip Ilic

Some Things

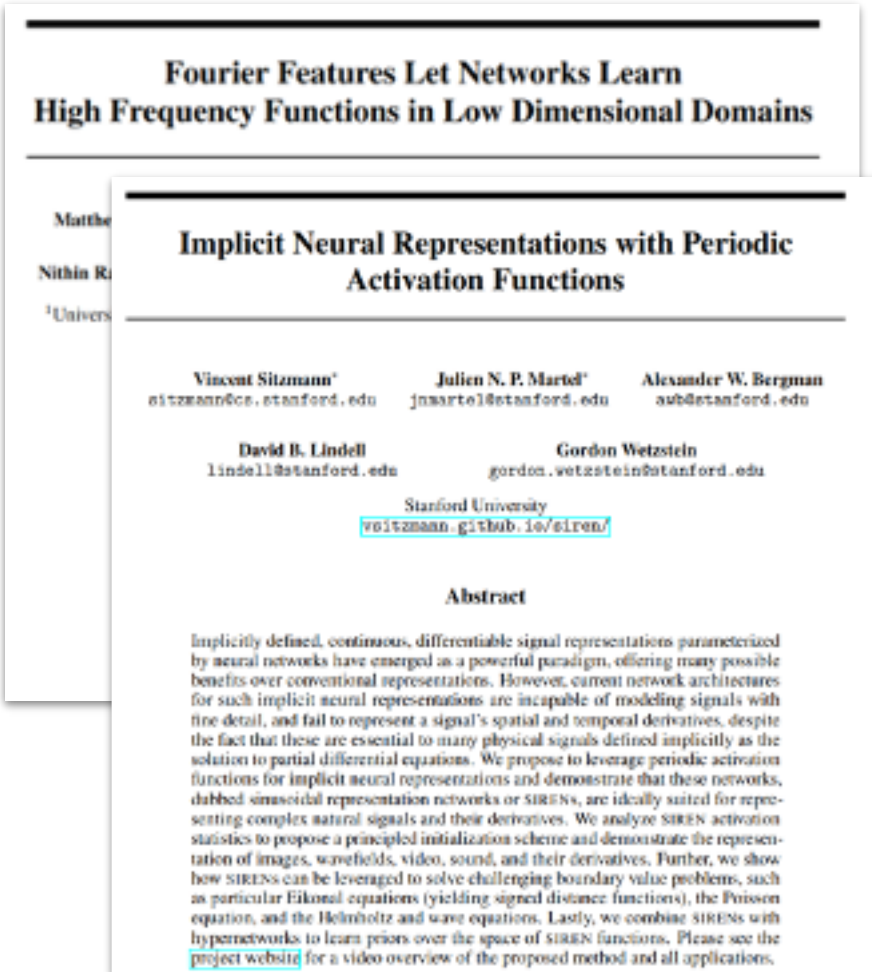
~~Everything~~ you wanted to know about NeRFs.

Reading Group, April 7th

Filip Ilic

Thing we are all familiar with

$x \rightarrow \Phi(x) =$
- Label
- Pose Estimation
- Style Transfer
- …



Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains

Implicit Neural Representations with Periodic Activation Functions

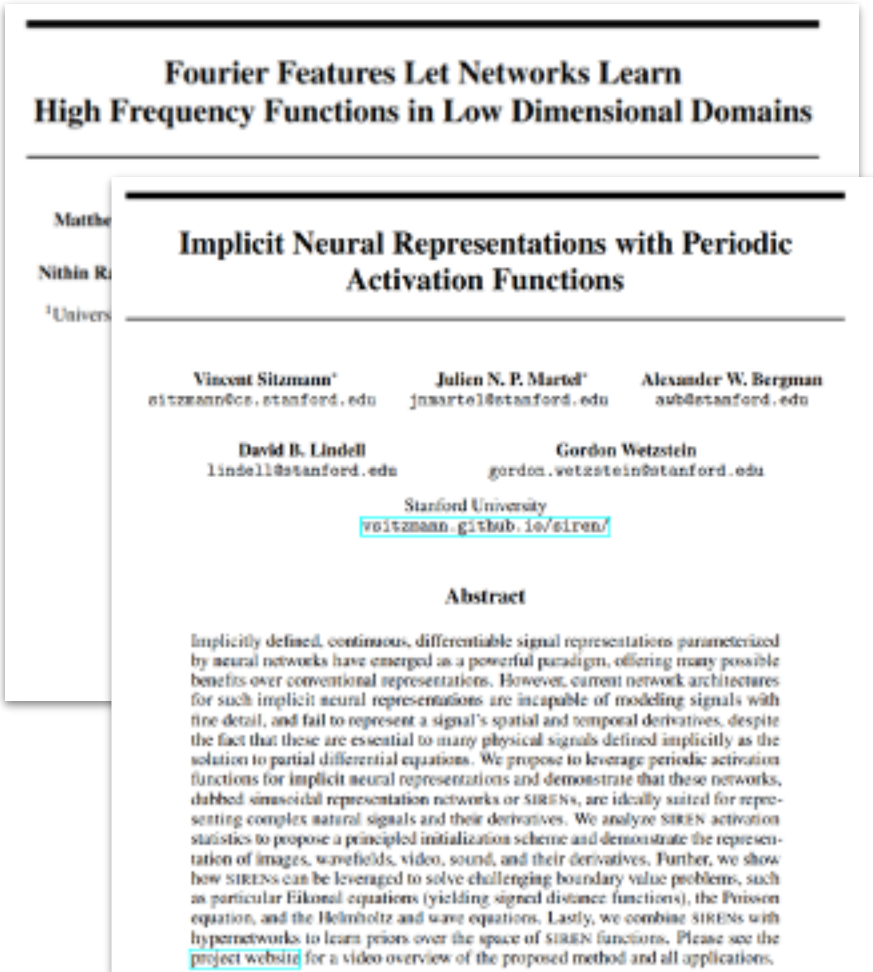
Vincent Sitzmann* sitzmann@cs.stanford.edu
Julien N. P. Martel* jnmartel@stanford.edu
Alexander W. Bergman awb@stanford.edu
David B. Lindell lindell@stanford.edu
Gordon Wetzstein gordon.wetzstein@stanford.edu
Stanford University
vsitzmann.github.io/siren/

Abstract

Implicitly defined, continuous, differentiable signal representations parameterized by neural networks have emerged as a powerful paradigm, offering many possible benefits over conventional representations. However, current network architectures for such implicit neural representations are incapable of modeling signals with fine detail, and fail to represent a signal's spatial and temporal derivatives, despite the fact that these are essential to many physical signals defined implicitly as the solution to partial differential equations. We propose to leverage periodic activation functions for implicit neural representations and demonstrate that these networks, dubbed sinusoidal representation networks or SIRENs, are ideally suited for representing complex natural signals and their derivatives. We analyze SIREN activation statistics to propose a principled initialization scheme and demonstrate the representation of images, wavefields, video, sound, and their derivatives. Further, we show how SIRENs can be leveraged to solve challenging boundary value problems, such as particular Eikonal equations (yielding signed distance functions), the Poisson equation, and the Helmholtz and wave equations. Lastly, we combine SIRENs with hypernetworks to learn priors over the space of SIREN functions. Please see the project website for a video overview of the proposed method and all applications.

$x \rightarrow \Phi(x) =$ Label
Pose Estimation
Style Transfer
…

Thing we are all familiar with



Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains

Matthe…

Nithin R…

[1]Univers…

Implicit Neural Representations with Periodic Activation Functions

Vincent Sitzmann[*] sitzmann@cs.stanford.edu

Julien N. P. Martel[*] jnmartel@stanford.edu

Alexander W. Bergman awb@stanford.edu

David B. Lindell lindell@stanford.edu

Gordon Wetzstein gordon.wetzstein@stanford.edu

Stanford University
vsitzmann.github.io/siren/

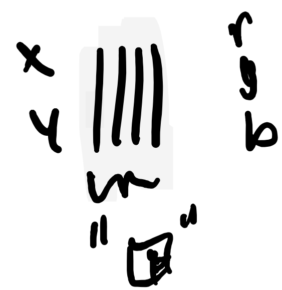
Abstract

Implicitly defined, continuous, differentiable signal representations parameterized by neural networks have emerged as a powerful paradigm, offering many possible benefits over conventional representations. However, current network architectures for such implicit neural representations are incapable of modeling signals with fine detail, and fail to represent a signal's spatial and temporal derivatives, despite the fact that these are essential to many physical signals defined implicitly as the solution to partial differential equations. We propose to leverage periodic activation functions for implicit neural representations and demonstrate that these networks, dubbed sinusoidal representation networks or SIRENs, are ideally suited for representing complex natural signals and their derivatives. We analyze SIREN activation statistics to propose a principled initialization scheme and demonstrate the representation of images, wavefields, video, sound, and their derivatives. Further, we show how SIRENs can be leveraged to solve challenging boundary value problems, such as particular Eikonal equations (yielding signed distance functions), the Poisson equation, and the Helmholtz and wave equations. Lastly, we combine SIRENs with hypernetworks to learn priors over the space of SIREN functions. Please see the project website for a video overview of the proposed method and all applications.

$$\Phi\begin{pmatrix}1\\2\end{pmatrix} = \begin{pmatrix}0.1\\0.2\\0.5\end{pmatrix}$$

$$\Phi\begin{pmatrix}1.2\\2.6\end{pmatrix} = \begin{pmatrix}\vdots\end{pmatrix}$$

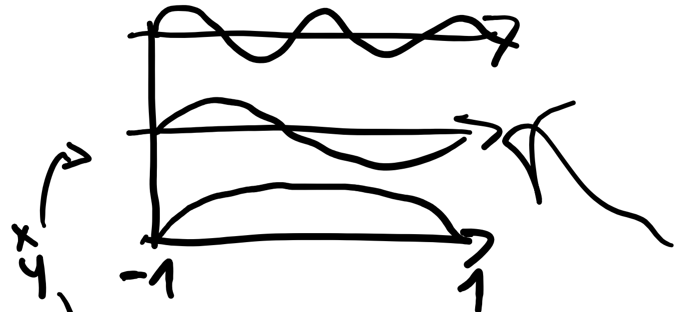
$$\begin{pmatrix}x\\y\end{pmatrix} \xrightarrow{\Phi} \begin{pmatrix}r\\g\\b\end{pmatrix}$$

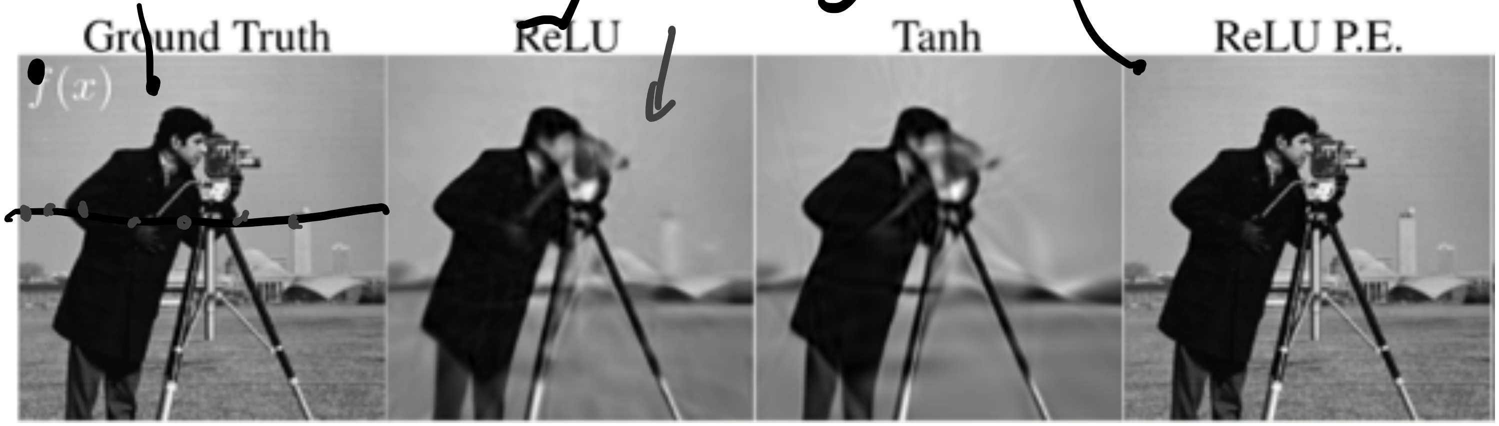


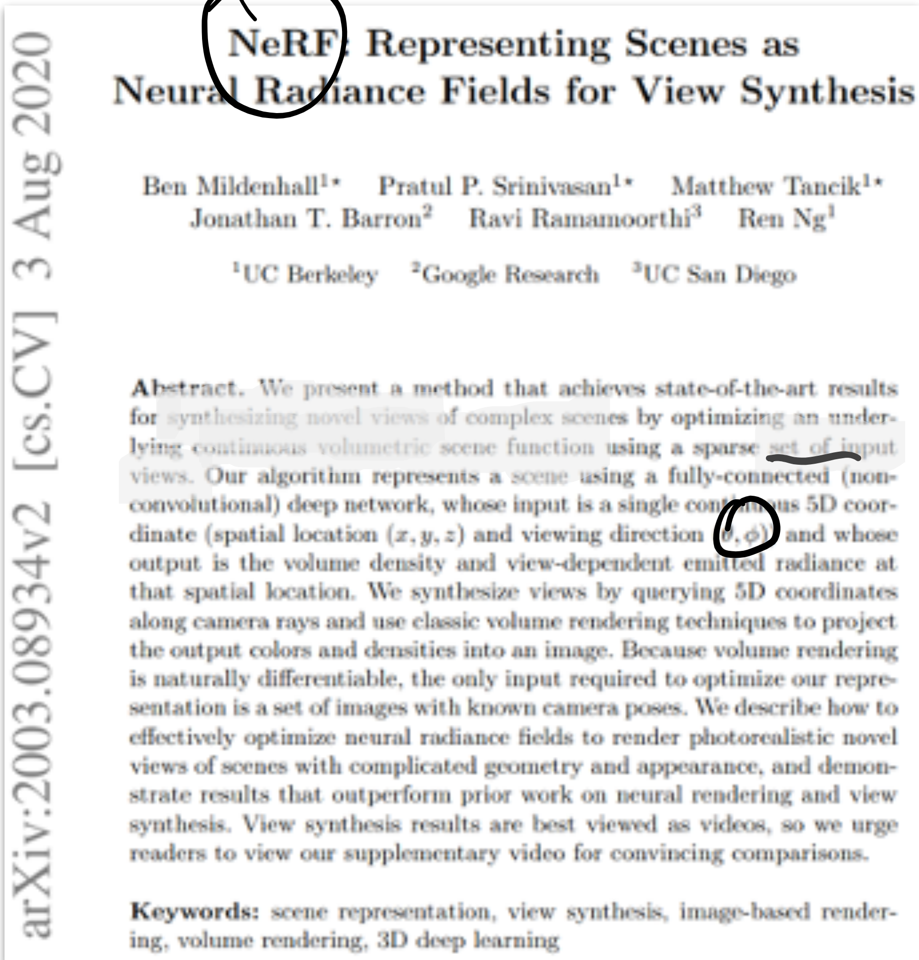
$x \rightarrow \begin{pmatrix} \sin(100x) \\ \sin(10x) \end{pmatrix}$

xy

gray







arXiv:2003.08934v2 [cs.CV] 3 Aug 2020

NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis

Ben Mildenhall[1*] Pratul P. Srinivasan[1*] Matthew Tancik[1*]
Jonathan T. Barron[2] Ravi Ramamoorthi[3] Ren Ng[1]

[1]UC Berkeley [2]Google Research [3]UC San Diego

Abstract. We present a method that achieves state-of-the-art results for synthesizing novel views of complex scenes by optimizing an underlying continuous volumetric scene function using a sparse set of input views. Our algorithm represents a scene using a fully-connected (non-convolutional) deep network, whose input is a single continuous 5D coordinate (spatial location (x, y, z) and viewing direction (θ, ϕ)) and whose output is the volume density and view-dependent emitted radiance at that spatial location. We synthesize views by querying 5D coordinates along camera rays and use classic volume rendering techniques to project the output colors and densities into an image. Because volume rendering is naturally differentiable, the only input required to optimize our representation is a set of images with known camera poses. We describe how to effectively optimize neural radiance fields to render photorealistic novel views of scenes with complicated geometry and appearance, and demonstrate results that outperform prior work on neural rendering and view synthesis. View synthesis results are best viewed as videos, so we urge readers to view our supplementary video for convincing comparisons.

Keywords: scene representation, view synthesis, image-based rendering, volume rendering, 3D deep learning

- Project Page: https://www.matthewtancik.com/nerf

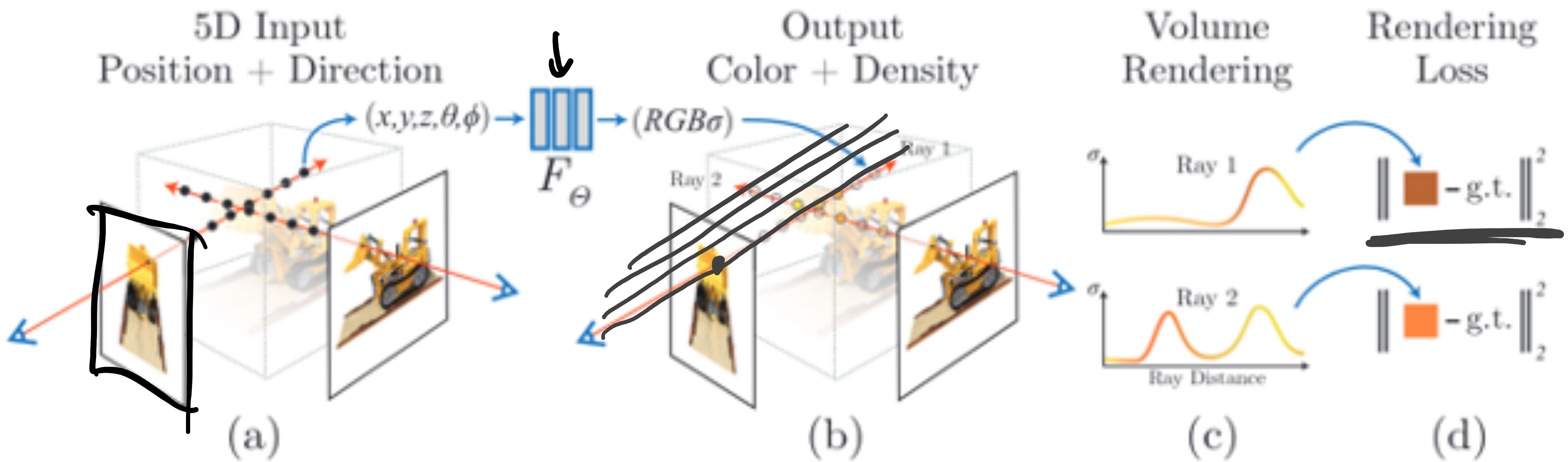
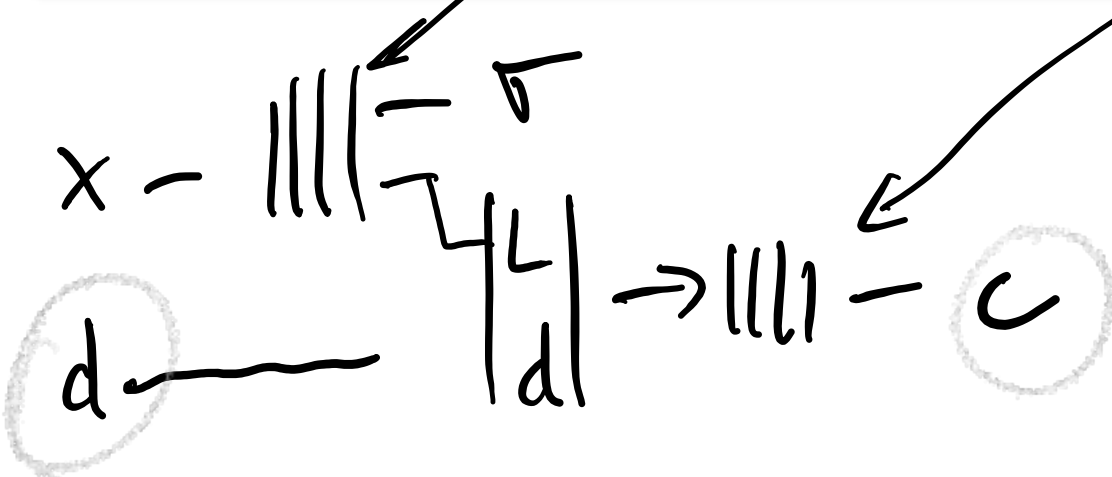


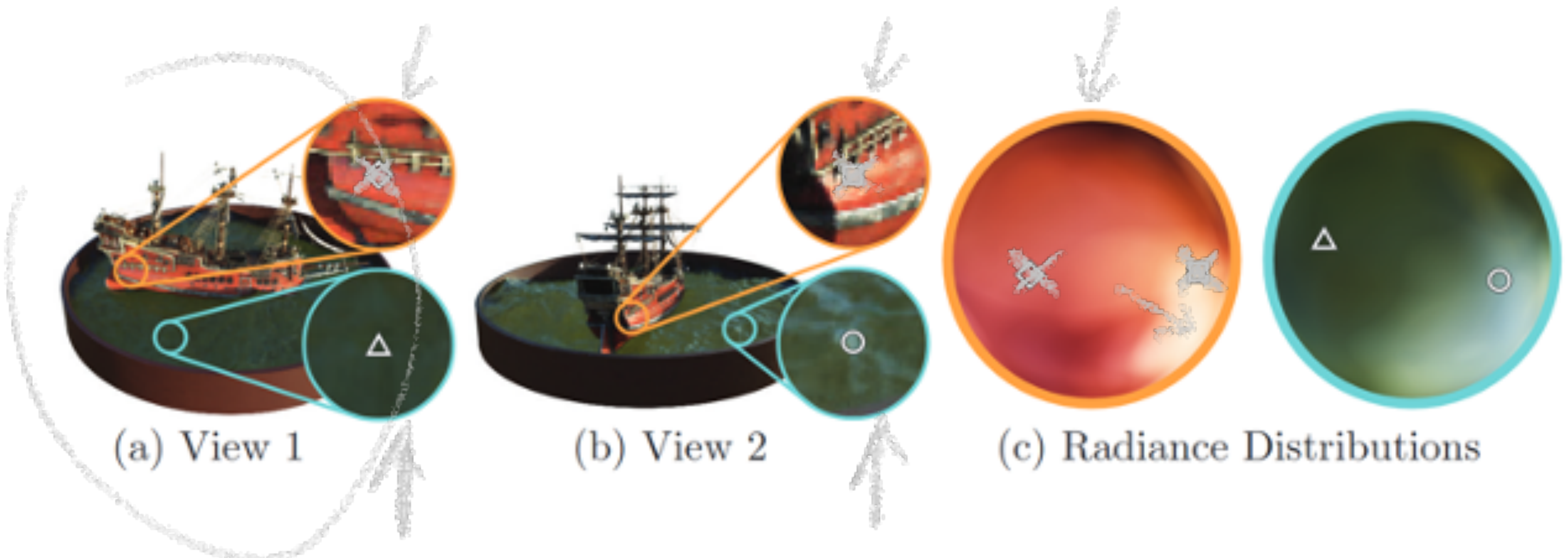
Fig. 2: An overview of our neural radiance field scene representation and differentiable rendering procedure. We synthesize images by sampling 5D coordinates (location and viewing direction) along camera rays (a), feeding those locations into an MLP to produce a color and volume density (b), and using volume rendering techniques to composite these values into an image (c). This rendering function is differentiable, so we can optimize our scene representation by minimizing the residual between synthesized and ground truth observed images (d).

Multiview consistency

We encourage the representation to be multiview consistent by restricting the network to predict the volume density σ as a function of only the location $\mathbf{x}$, while allowing the RGB color $\mathbf{c}$ to be predicted as a function of both location and viewing direction. To accomplish this, the MLP F_Θ first processes the input 3D coordinate $\mathbf{x}$ with 8 fully-connected layers (using ReLU activations and 256 channels per layer), and outputs σ and a 256-dimensional feature vector. This feature vector is then concatenated with the camera ray's viewing direction and passed to one additional fully-connected layer (using a ReLU activation and 128 channels) that output the view-dependent RGB color.



Multiview consistency - view dependent emitted radiance

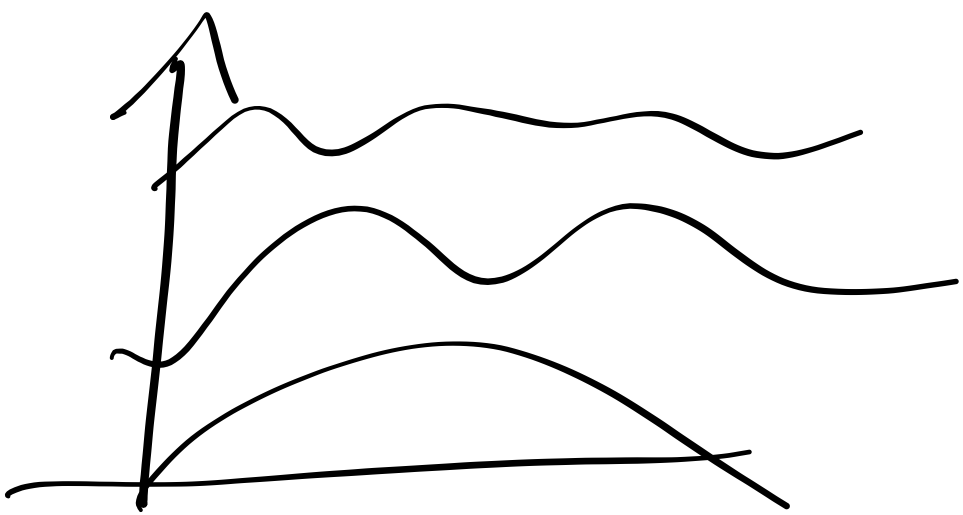


Positional Encoding

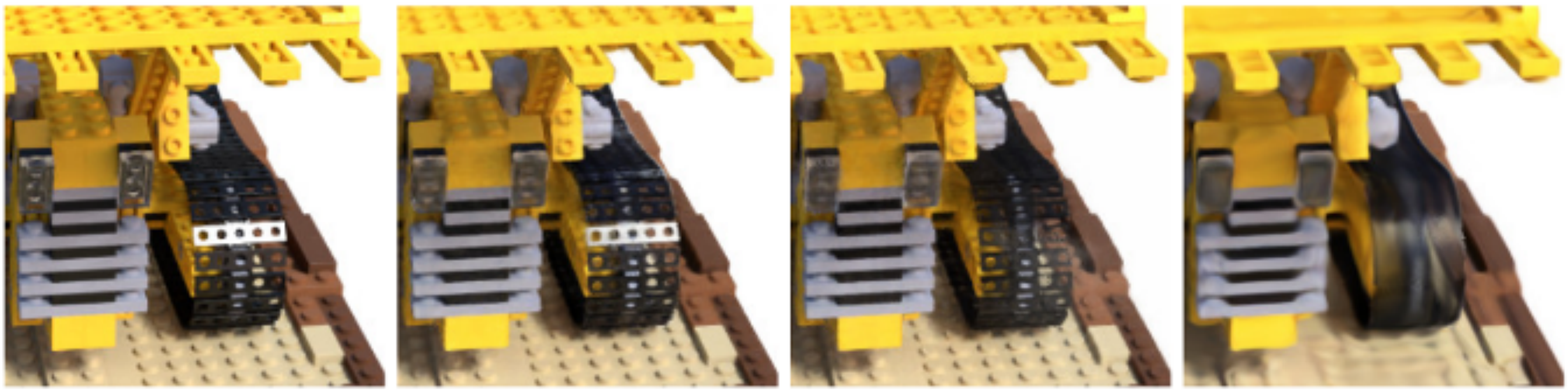
Despite the fact that neural networks are universal function approximators [14], we found that having the network F_Θ directly operate on $xyz\theta\phi$ input coordinates results in renderings that perform poorly at representing high-frequency variation in color and geometry. This is consistent with recent work by Rahaman *et al.* [35], which shows that deep networks are biased towards learning lower frequency functions. They additionally show that mapping the inputs to a higher dimensional space using high frequency functions before passing them to the network enables better fitting of data that contains high frequency variation.

We leverage these findings in the context of neural scene representations, and show that reformulating F_Θ as a composition of two functions $F_\Theta = F'_\Theta \circ \gamma$, one learned and one not, significantly improves performance (see Fig. 4 and Table 2). Here γ is a mapping from $\mathbb{R}$ into a higher dimensional space $\mathbb{R}^{2L}$, and F'_Θ is still simply a regular MLP. Formally, the encoding function we use is:

$$\gamma(p) = \left(\sin(2^0\pi p), \cos(2^0\pi p), \cdots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p)\right). \quad (4)$$



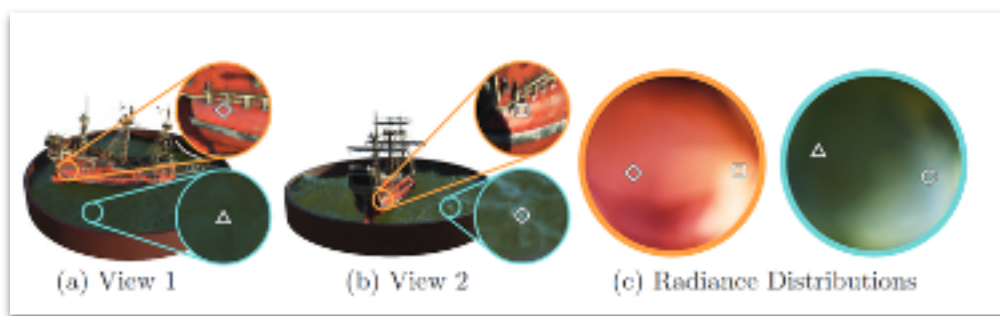
Positional Encoding / View dependence



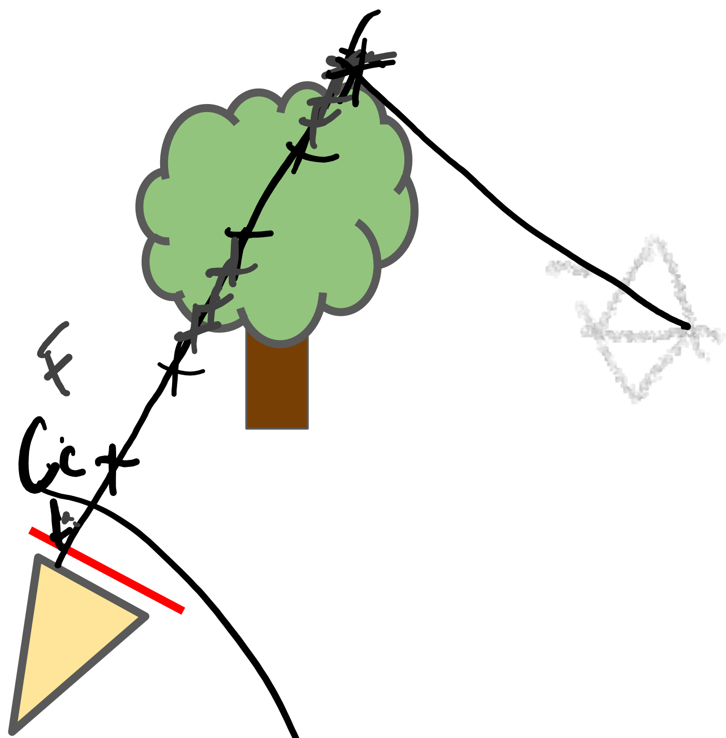
Ground Truth | Complete Model | No View Dependence | No Positional Encoding

Lambertian

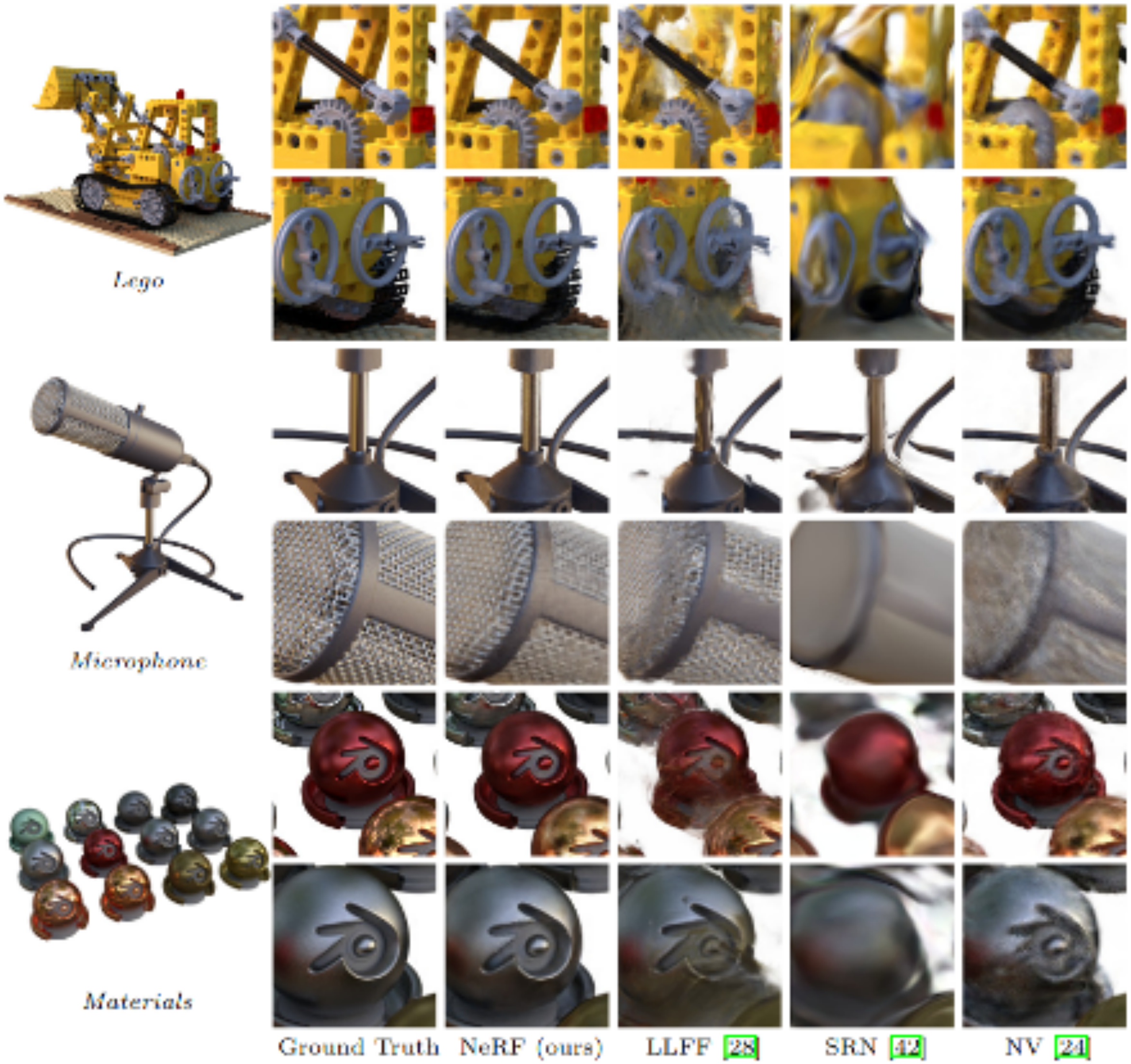
- https://www.matthewtancik.com/nerf



Coarse + Fine Model



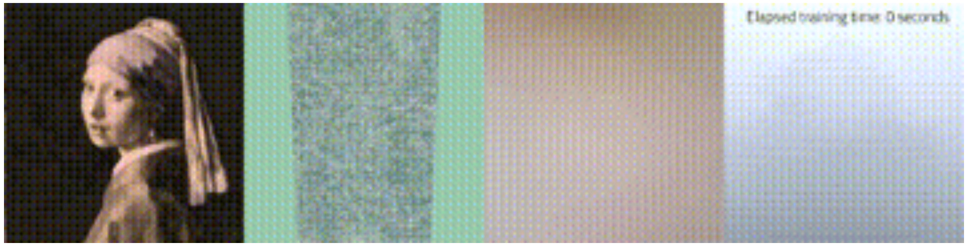
$$\mathcal{L} = \sum_{\mathbf{r} \in \mathcal{R}} \left[\left\| \hat{C}_c(\mathbf{r}) - C(\mathbf{r}) \right\|_2^2 + \left\| \hat{C}_f(\mathbf{r}) - C(\mathbf{r}) \right\|_2^2 \right]$$



Continue from here:
https://docs.google.com/presentation/d/1ZihN67cNRbjs6xbEcbWSOypQhZtAa6XvssaIBrVyiZ0/edit#slide=id.g1236fe90073_0_0

Possible Issues?

- Slow to train / inference: Nvidia instant ngp, trains in seconds.



- What happens if the images are not of a stationary subject?

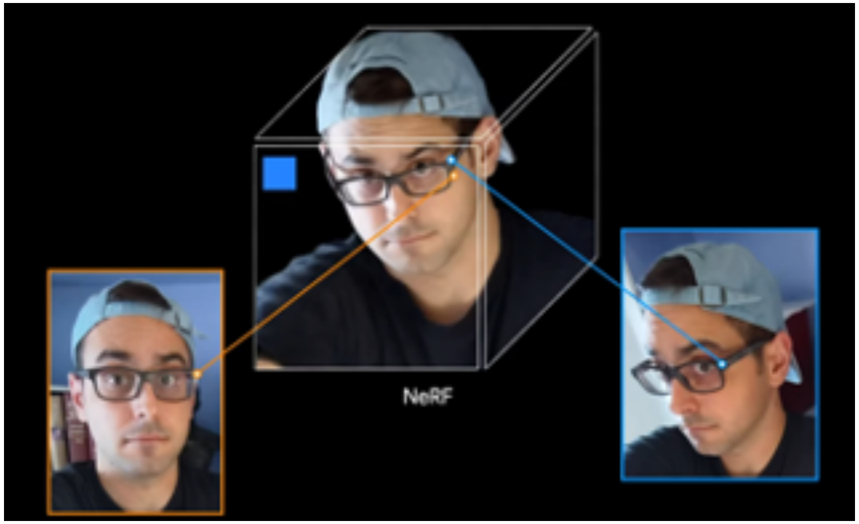
NeRF →

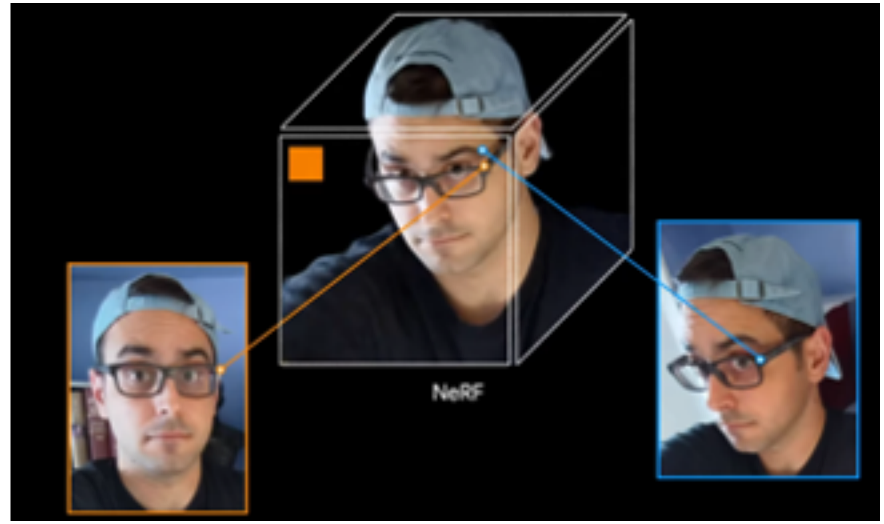
Nerfies: Deformable Neural Radiance Fields

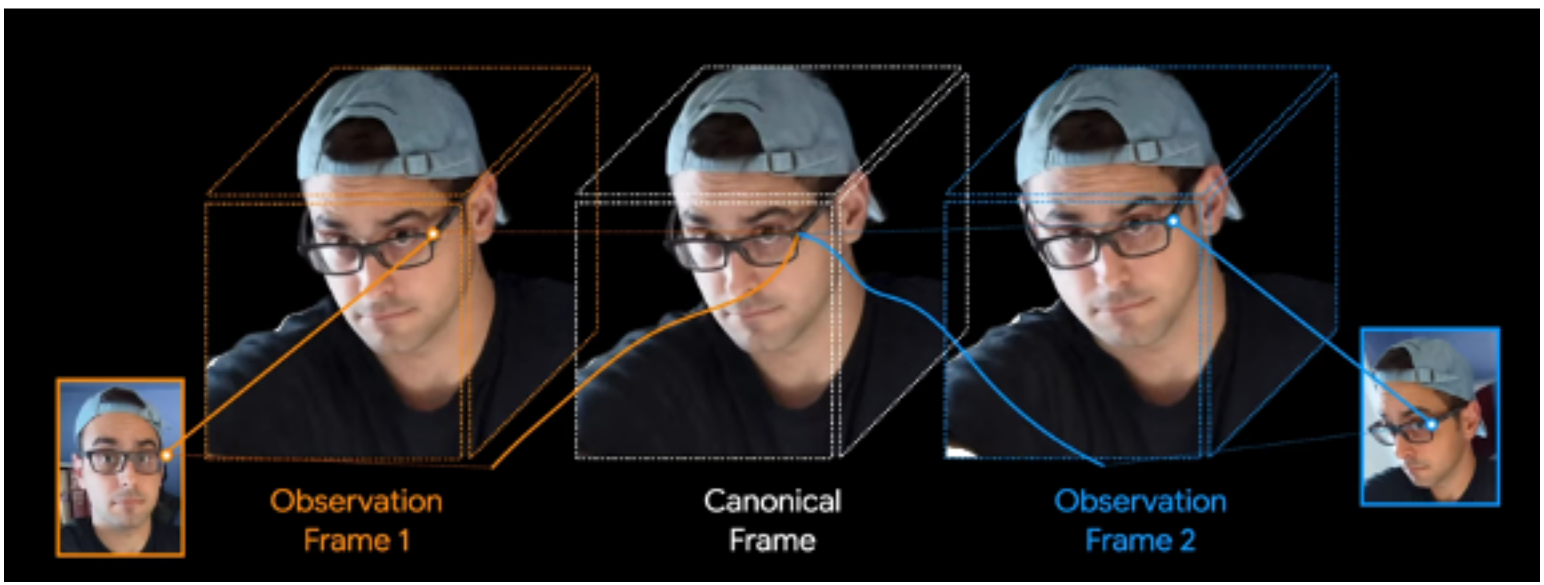
Keunhong Park[1*] Utkarsh Sinha[2] Jonathan T. Barron[2] Sofien Bouaziz[2]
Dan B Goldman[2] Steven M. Seitz[1,2] Ricardo Martin-Brualla[2]
[1]University of Washington [2]Google Research
nerfies.github.io

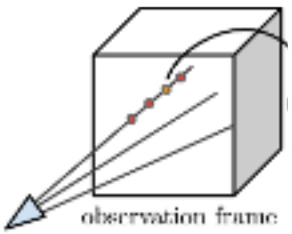
(a) casual capture (b) input images (c) nerfie novel views (d) novel view depth

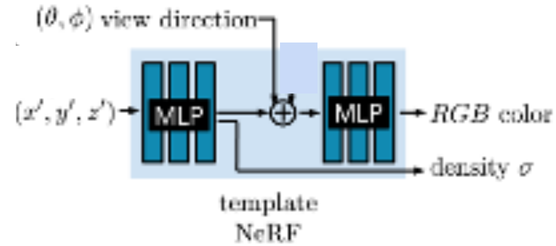
Figure 1: We reconstruct photo-realistic *nerfies* from a user casually waving a mobile phone (a). Our system uses selfie photos/videos (b) to produce a free-viewpoint representation with accurate renders (c) and geometry (d). Please see video.

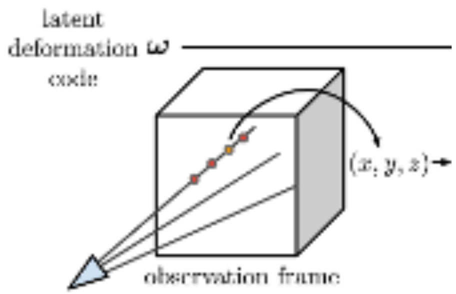

NeRF

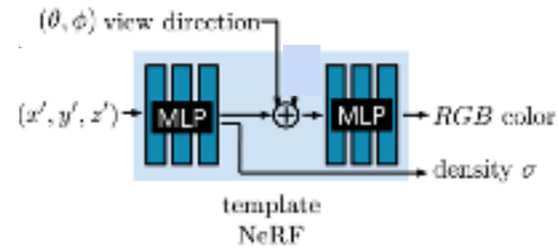

NeRF


Observation
Frame 1
Canonical
Frame
Observation
Frame 2

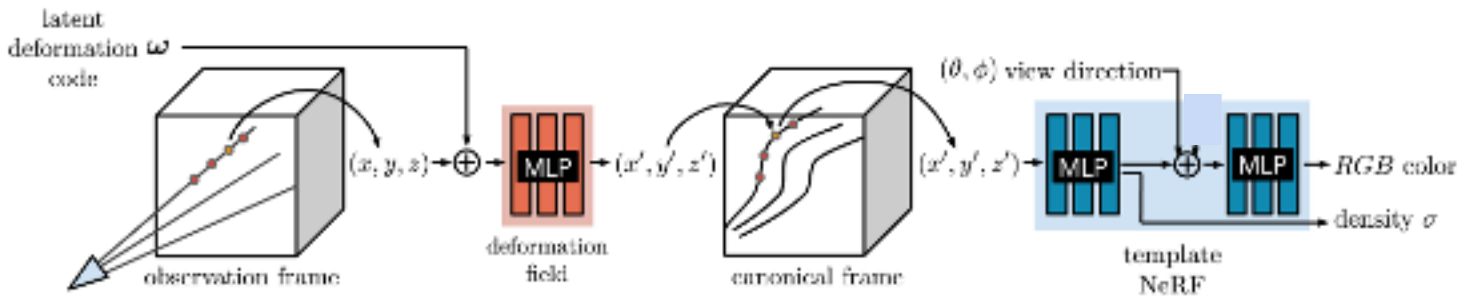

observation frame


(θ, ϕ) view direction
(x', y', z')
MLP
MLP
RGB color
density σ
template
NeRF


latent
deformation ω
code
(x, y, z)
observation frame


(θ, ϕ) view direction
(x', y', z')
MLP
MLP
RGB color
density σ
template
NeRF


latent
deformation ω
code
(x, y, z)
MLP
deformation
field
observation frame
(θ, ϕ) view direction
(x', y', z')
MLP
MLP
RGB color
density σ
template
NeRF


latent
deformation ω
code
(x, y, z)
MLP
deformation
field
(x', y', z')
observation frame
canonical frame
(θ, ϕ) view direction
(x', y', z')
MLP
MLP
RGB color
density σ
template
NeRF

reg

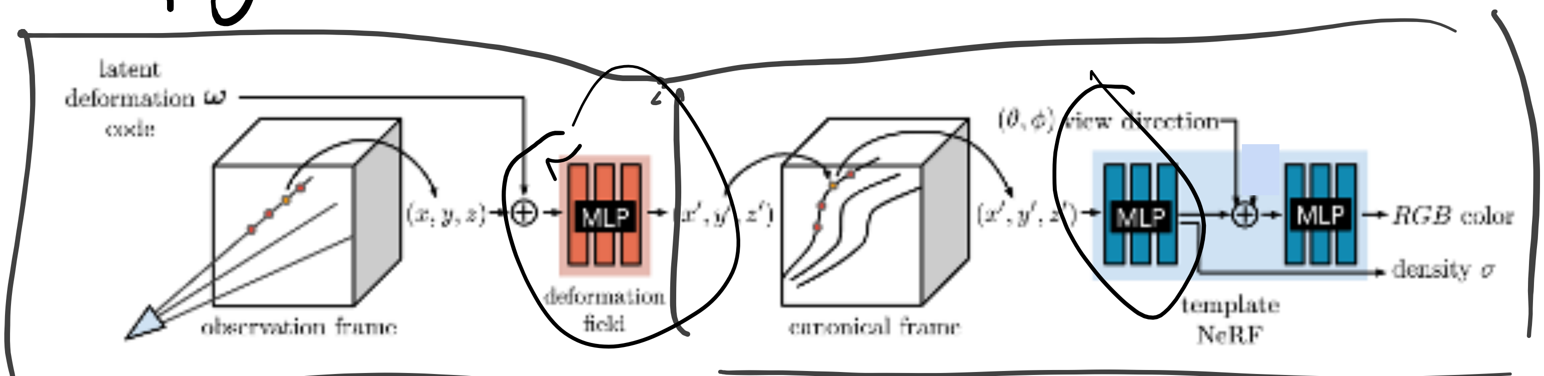


Figure 2: We associate a latent deformation code (ω) and an appearance code (ψ) to each image. We trace the camera rays in the observation frame and transform samples along the ray to the canonical frame using a deformation field encoded as an MLP that is conditioned on the deformation code ω. We query the template NeRF [39] using the transformed sample (x', y', z'), viewing direction (θ, ϕ) and appearance code ψ as inputs to the MLP and integrate samples along the ray following NeRF.

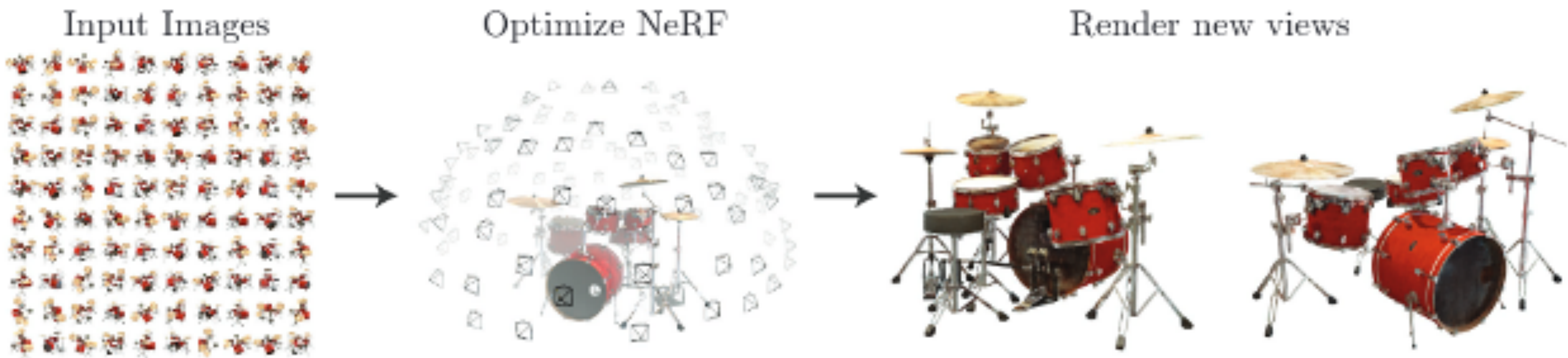
Concluding: Lets take a look at some nerfies!
https://nerfies.github.io/

Talks, Literature, Project pages

- Papers:
  - Nerf: https://arxiv.org/pdf/2003.08934.pdf
  - Implicit Neural Representations with Periodic
  - Activation Functions: https://arxiv.org/pdf/2006.09661.pdf
  - Nerfies: Deformable Radiance Fields https://arxiv.org/pdf/2011.12948.pdf
  - Instant ngp: https://nvlabs.github.io/instant-ngp/assets/mueller2022instant.pdf\

- 1h talk of author: https://www.youtube.com/watch?v=HfJpQCBTqZs
- Nerf Github page: https://github.com/bmild/nerf
- Nerfies Spotlight: https://www.youtube.com/watch?v=MrKrnHhk8IA
- Nerfies Github: https://github.com/google/nerfies
- Instant ngp: https://nvlabs.github.io/instant-ngp/

Backup slides



- Structure from motion
- SLAM
- …