Neuralangelo: High-Fidelity Neural Surface Reconstruction

Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H. Taylor, Mathias Unberath, Ming-Yu Liu, Chen-Hsuan Lin
NVIDIA Research, Johns Hopkins University

Outline

- Introduction
- Citations and Related Works
- Preliminaries
- Proposed Method
- Optimization
- Results

Introduction

Neuralangelo is a framework for high-fidelity 3D surface reconstruction from RGB images using neural volume rendering, without using auxiliary data such as segmentation or depth.

Citations and Related Works

- NeuS2: Fast Learning of Neural Implicit Surfaces for Multi-view Reconstruction
  - Yiming Wang et al. (ICCV 2023)

- SuGaR: Surface-Aligned Gaussian Splatting for Efficient 3D Mesh Reconstruction and High-Quality Mesh Rendering
  - Antoine Guédon, Vincent Lepetit (CVPR 2024)

- GaussianEditor: Swift and Controllable 3D Editing with Gaussian Splatting
  - Yiwen Chen et al. (CVPR 2024)


Neuralangelo: High-Fidelity Neural Surface Reconstruction
Zhaoshuo Li[1,2], Thomas Müller[1], Alex Evans[1], Russell H. Taylor[2], Mathias Unberath[2], Ming-Yu Liu[1], Chen-Hsuan Lin[1]
[1]NVIDIA Research [2]Johns Hopkins University

Preliminaries (Neural Volume Rendering)

$$\hat{\mathbf{c}}(\mathbf{o},\mathbf{d}) = \sum_{i=1}^{N} w_i \mathbf{c}_i, \quad \text{where } w_i = T_i \alpha_i$$

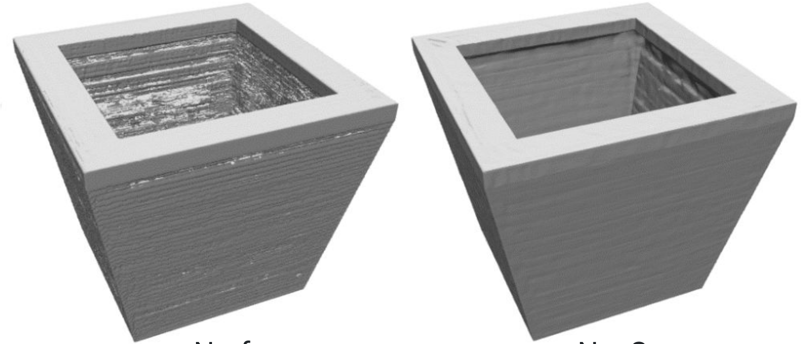
$\alpha_i = 1 - \exp(-\sigma_i \delta_i)$ is the opacity of the ith segment.

$\delta_i = t_{i+1} - t_i$ is the distance between adjacent samples.

$T_i = \Pi_{j=1}^{i-1}(1 - \alpha_j)$ is the accumulated transmittance.

NeRF (B. Mildenhall, et al - ECCV 2020)

Preliminaries (NeuS)



Nerf NeuS

$$\hat{\mathbf{c}}(\mathbf{o},\mathbf{d}) = \sum_{i=1}^{N} w_i \mathbf{c}_i, \quad \text{where } w_i = T_i \alpha_i$$

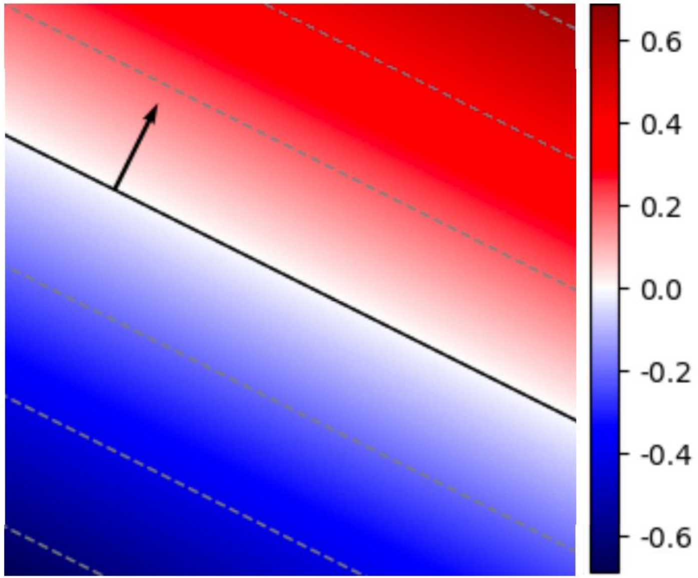
$\alpha_i = \max\left(\frac{\Phi_s(f(\mathbf{x}_i)) - \Phi_s(f(\mathbf{x}_{i+1}))}{\Phi_s(f(\mathbf{x}_i))}, 0\right)$ is the opacity of the i^{th} segment.

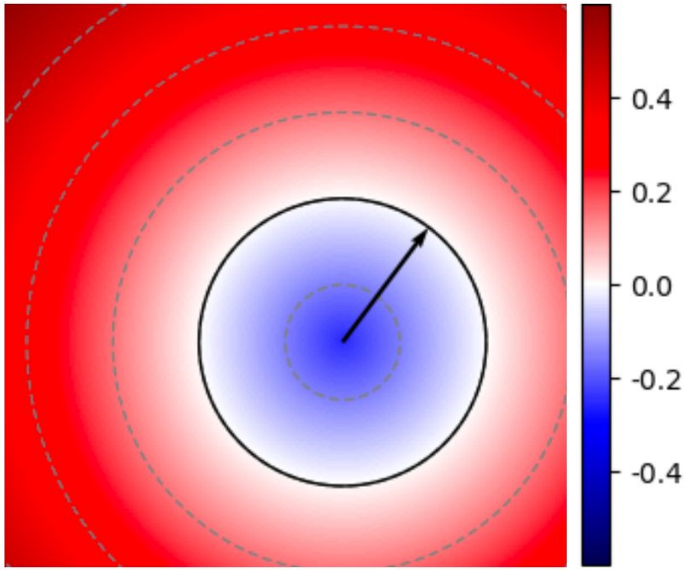
$T_i = \Pi_{j=1}^{i-1}(1-\alpha_j)$ is the accumulated transmittance.

NeuS (P. Wang, et al - NeurIPS 2021(Spotlight))

Signed Distance Function (SDF)

$$\phi(\mathbf{x}) = \min_{\mathbf{s} \in \mathcal{S}} \text{sign}(\mathbf{x}) \cdot \|\mathbf{x} - \mathbf{s}\|_2$$



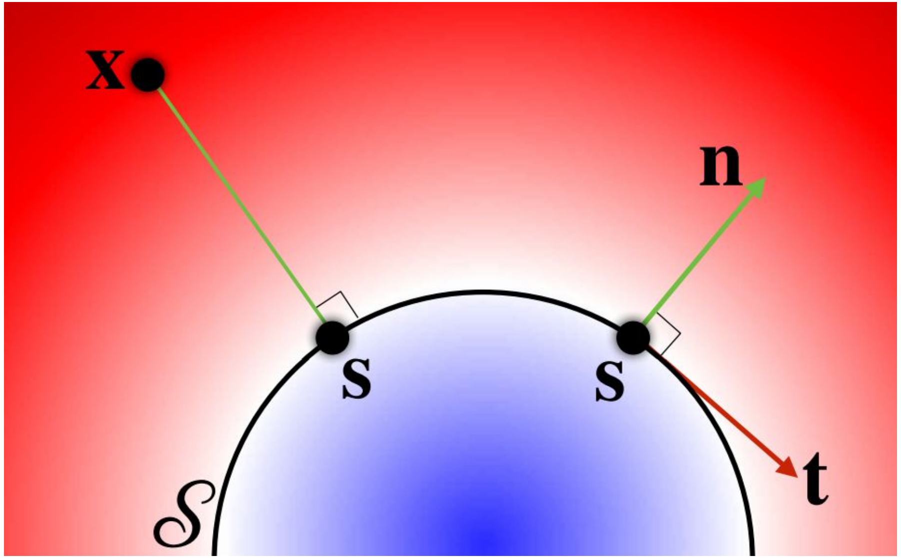


Properties of SDF

Surface Normals: $\mathbf{n} = \nabla \phi(\mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}$

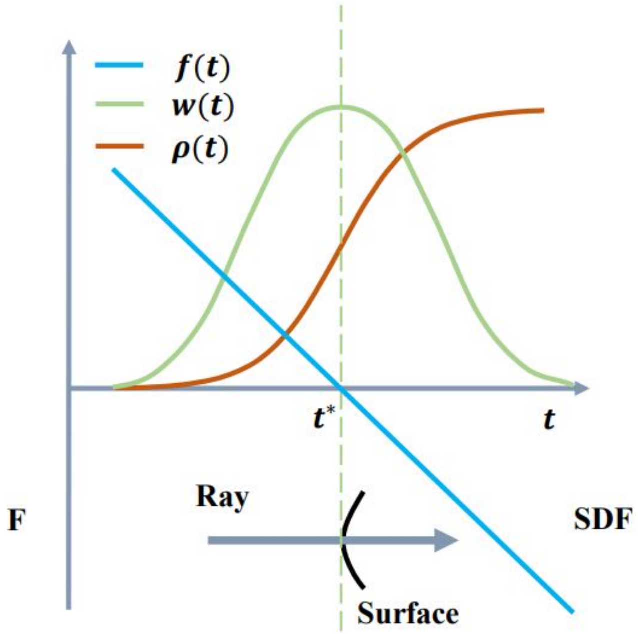
Eikonal Constraint: $\|\nabla \phi(\mathbf{x})\|_2 \equiv 1 \quad \forall \mathbf{x}$ (almost)

Surface Projection: $\mathbf{s} = \Pi_{\mathcal{S}}(\mathbf{x}) = -\nabla \phi(\mathbf{x}) \cdot \|\phi(\mathbf{x})\|$



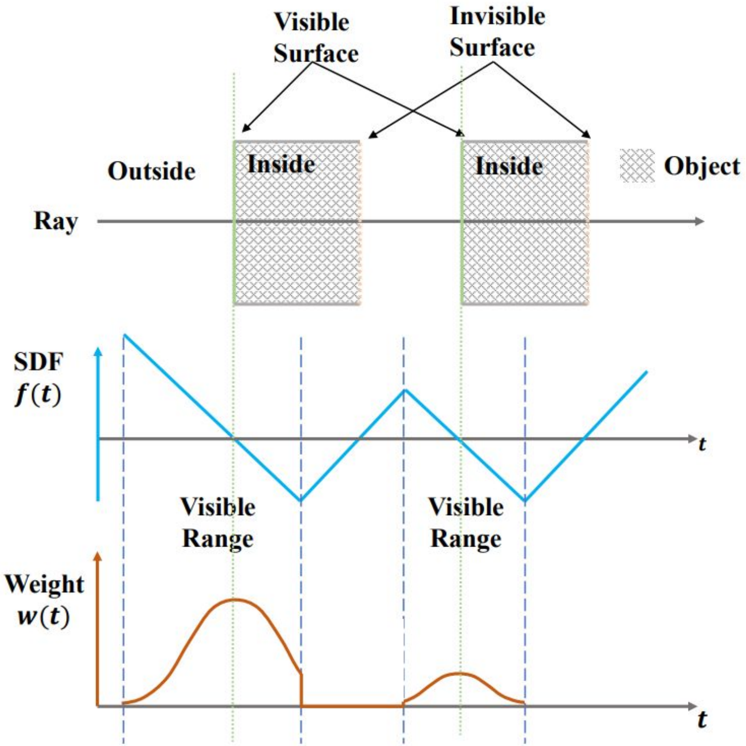
Requirements on Weight Function

Unbiased: Given a camera ray p(t), w(t) attains a locally maximal value at a surface intersection point p(t^*) (on the zero-level set of the SDF(x)).



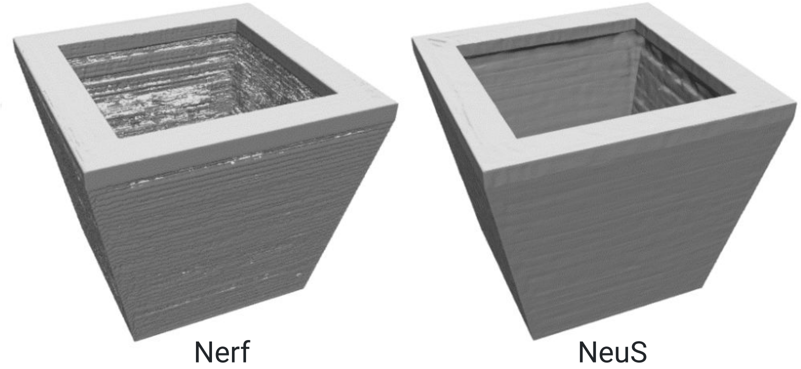
Requirements on Weight Function

Occlusion-aware: When two points have the same SDF value, the point nearer to the viewpoint should have a larger contribution to the final output color than does the other point.



Preliminaries (NeuS)

$$\hat{\mathbf{c}}(\mathbf{o},\mathbf{d}) = \sum_{i=1}^{N} w_i \mathbf{c}_i, \quad \text{where } w_i = T_i \alpha_i$$



$\alpha_i = \max\left(\frac{\Phi_s(f(\mathbf{x}_i)) - \Phi_s(f(\mathbf{x}_{i+1}))}{\Phi_s(f(\mathbf{x}_i))}, 0\right)$ is the opacity of the i^{th} segment.

$T_i = \Pi_{j=1}^{i-1}(1-\alpha_j)$ is the accumulated transmittance.

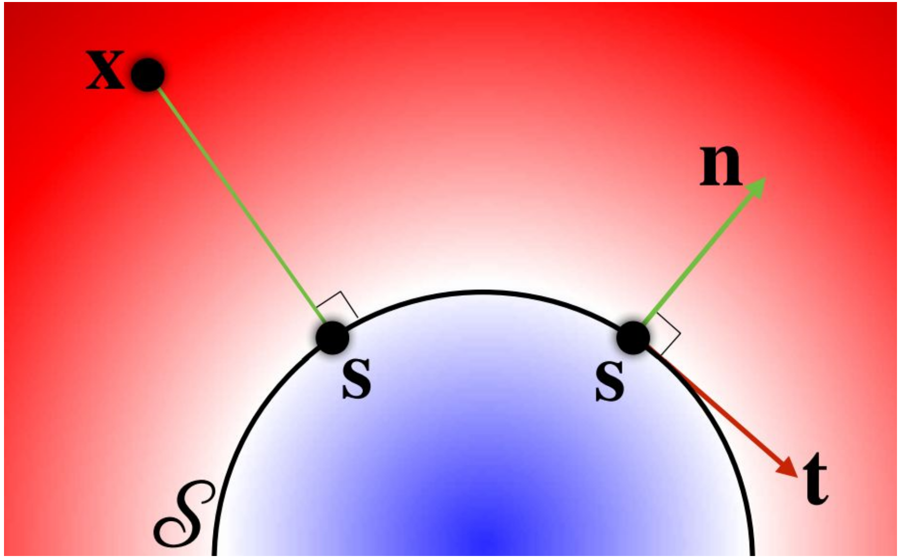
NeuS (P. Wang, et al - NeurIPS 2021(Spotlight))

Can we use other properties of SDF in our modelling?

Surface Normals: $\mathbf{n} = \nabla \phi(\mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}$

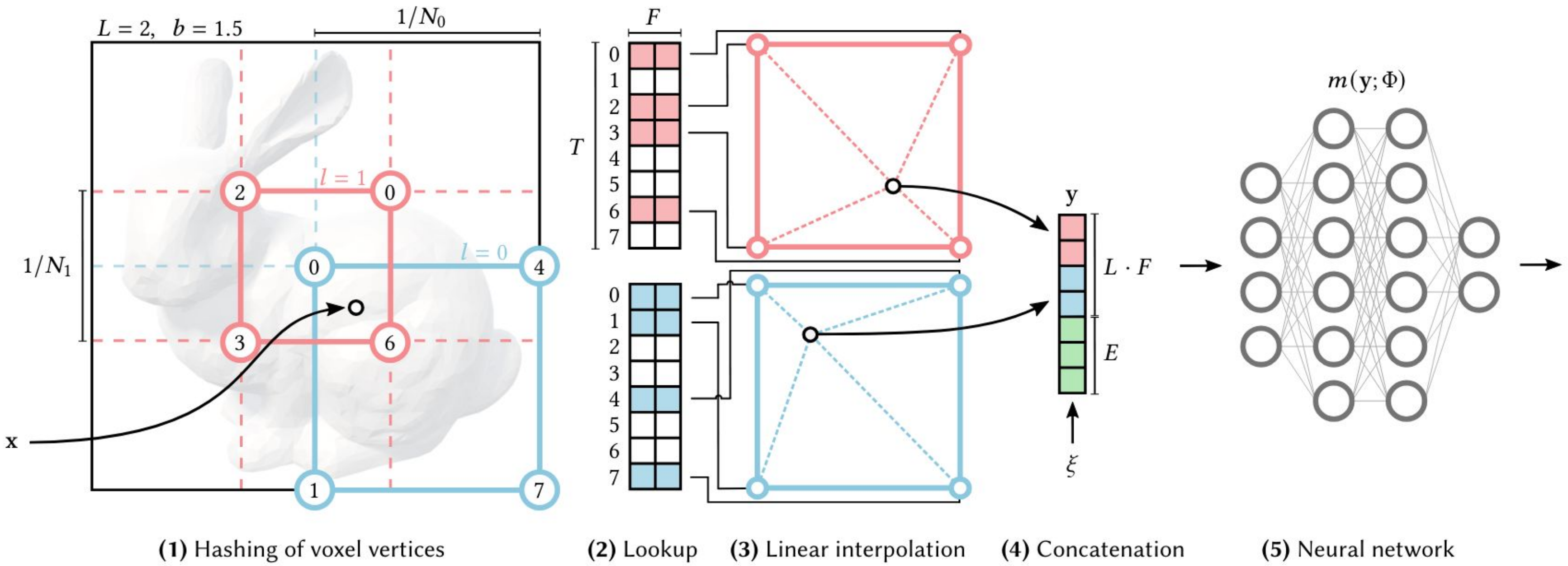
Eikonal Constraint: $\|\nabla \phi(\mathbf{x})\|_2 \equiv 1 \quad \forall \mathbf{x}$ (almost)

Surface Projection: $\mathbf{s} = \Pi_{\mathcal{S}}(\mathbf{x}) = -\nabla \phi(\mathbf{x}) \cdot \|\phi(\mathbf{x})\|$



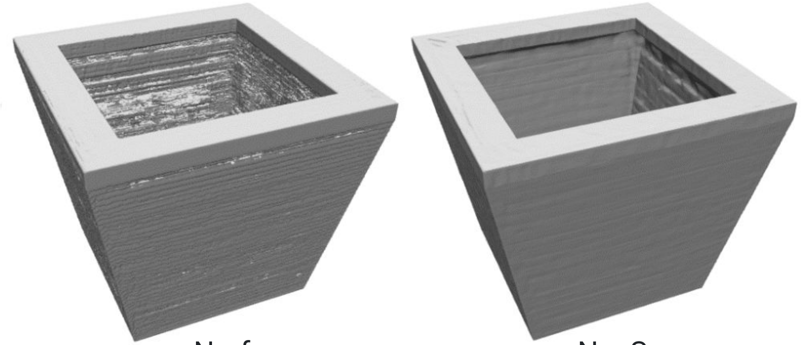
Preliminaries (Instant-NGP)

Key idea: multi-resolution hash encoding



Instant Neural Graphics Primitives with a Multiresolution Hash Encoding (T. Müller, et al -SIGGRAPH 2022)

Preliminaries (NeuS)



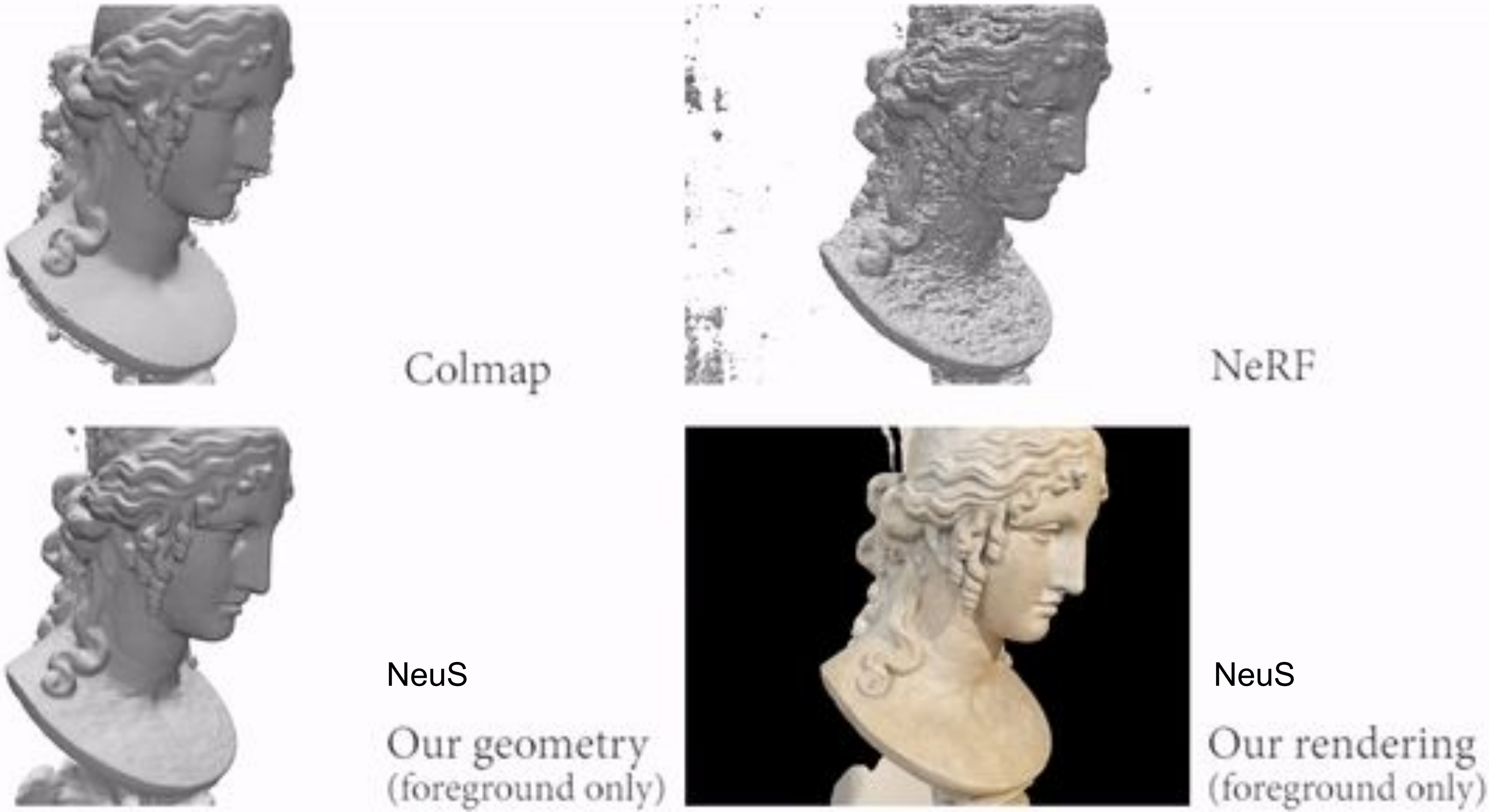
$$\hat{\mathbf{c}}(\mathbf{o}, \mathbf{d}) = \sum_{i=1}^{N} w_i \mathbf{c}_i, \quad \text{where } w_i = T_i \alpha_i$$

$\alpha_i = \max\left(\frac{\Phi_s(f(\mathbf{x}_i)) - \Phi_s(f(\mathbf{x}_{i+1}))}{\Phi_s(f(\mathbf{x}_i))}, 0\right)$ is the opacity of the i^{th} segment.

$T_i = \Pi_{j=1}^{i-1}(1 - \alpha_j)$ is the accumulated transmittance.

NeuS (P. Wang, et al - NeurIPS 2021(Spotlight))

Comparing NeuS with NeRF



Proposed Method

Two key ideas:

1. Numerical Gradient

2. Coarse-to-fine Optimization

Johns Hopkins University

Computing Surface Normals

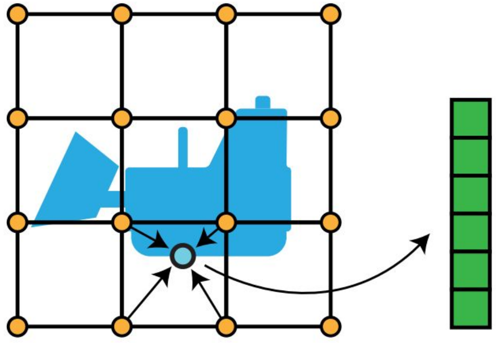
$$\gamma_l(\mathbf{x}_{i,l}) = \gamma_l(\lfloor \mathbf{x}_{i,l} \rfloor) \cdot (1 - \beta) + \gamma_l(\lceil \mathbf{x}_{i,l} \rceil) \cdot \beta$$

$\mathbf{x}_{i,l} = \mathbf{x}_i \cdot V_l$ is scaled 3D point by the grid resolution.

$\beta = \mathbf{x}_{i,l} - \lfloor \mathbf{x}_{i,l} \rfloor$ is the coefficient for (tri-)linear interpolation.

Therefore, the derivative of the hash encoding is

$$\frac{\partial \gamma_l(\mathbf{x}_{i,l})}{\partial \mathbf{x}_i} = \gamma_l(\lfloor \mathbf{x}_{i,l} \rfloor) \cdot (-\frac{\partial \beta}{\partial \mathbf{x}_i}) + \gamma_l(\lceil \mathbf{x}_{i,l} \rceil) \cdot \frac{\partial \beta}{\partial \mathbf{x}_i}$$
$$= \gamma_l(\lfloor \mathbf{x}_{i,l} \rfloor) \cdot (-V_l) + \gamma_l(\lceil \mathbf{x}_{i,l} \rceil) \cdot V_l \ .$$



Feature Grid Interpolation

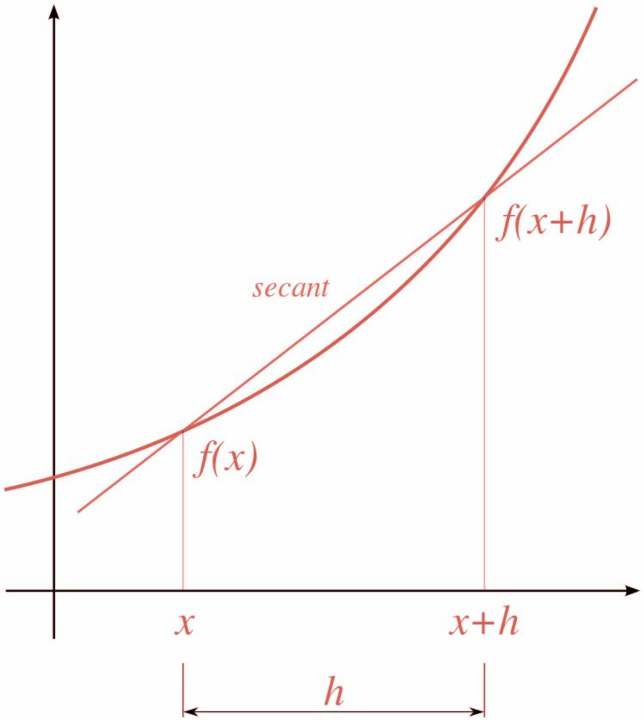
Numerical Gradient

Additional SDF samples are needed.

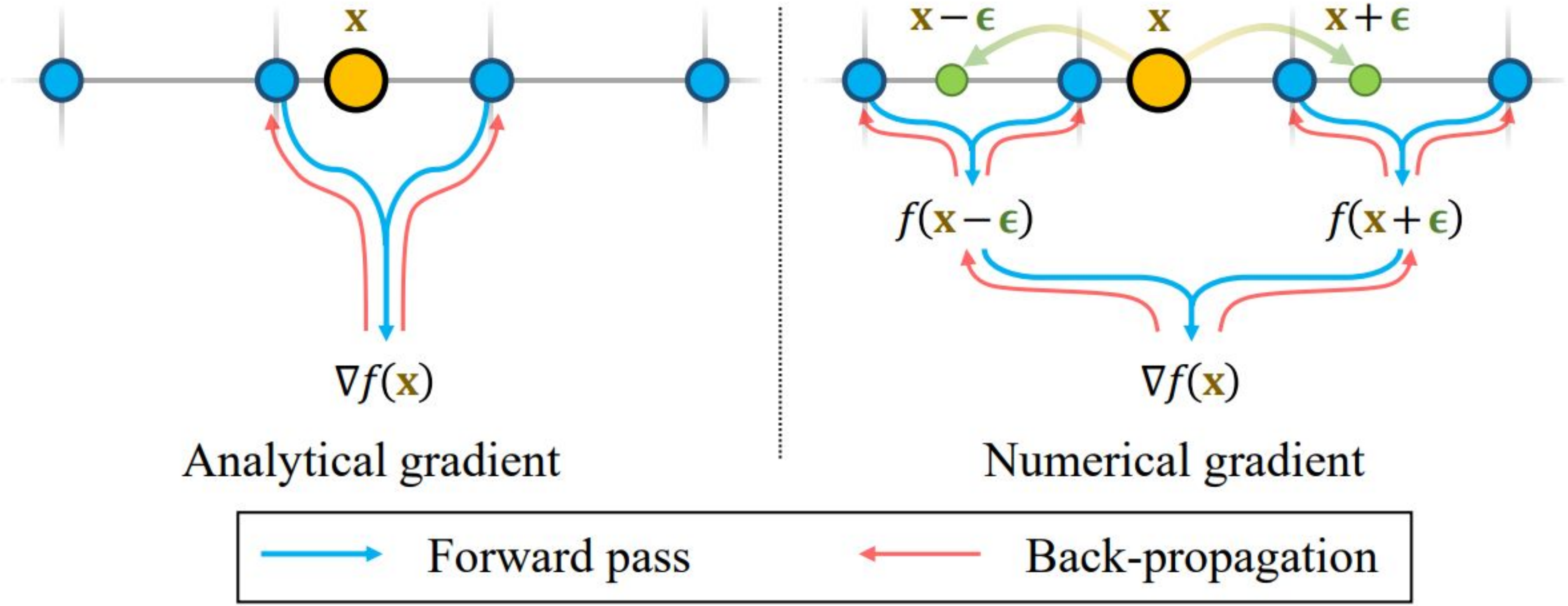
$$\nabla_x f(\mathbf{x}_i) = \frac{f\left(\gamma(\mathbf{x}_i + \boldsymbol{\epsilon}_x)\right) - f\left(\gamma(\mathbf{x}_i - \boldsymbol{\epsilon}_x)\right)}{2\epsilon}$$

where $\boldsymbol{\epsilon}_x = [\epsilon, 0, 0]$.

6 additional SDF samples are required.



Numerical Gradients

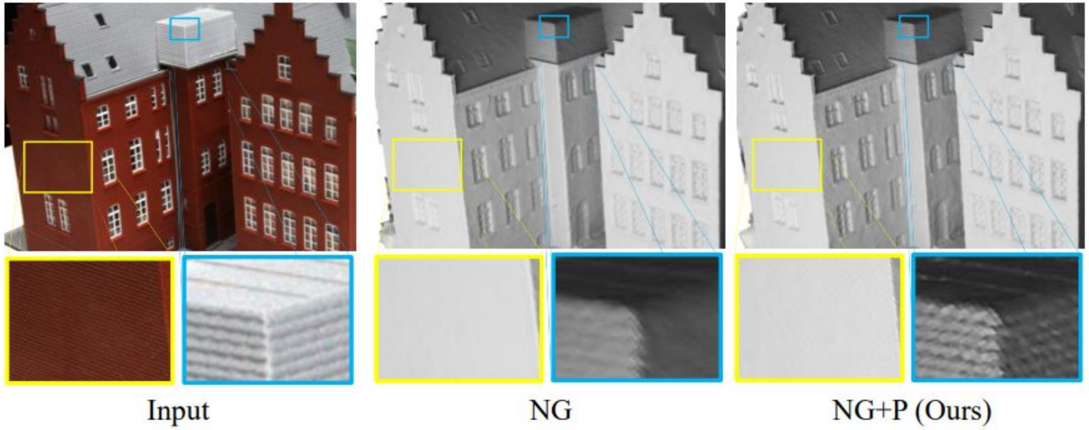


Progressive Levels of Details (Step size ϵ)

Role of ϵ: Controls the resolution and amount of recovered details in numerical gradients.

Large ϵ: Ensures surface normals are consistent at a larger scale, leading to continuous surfaces.

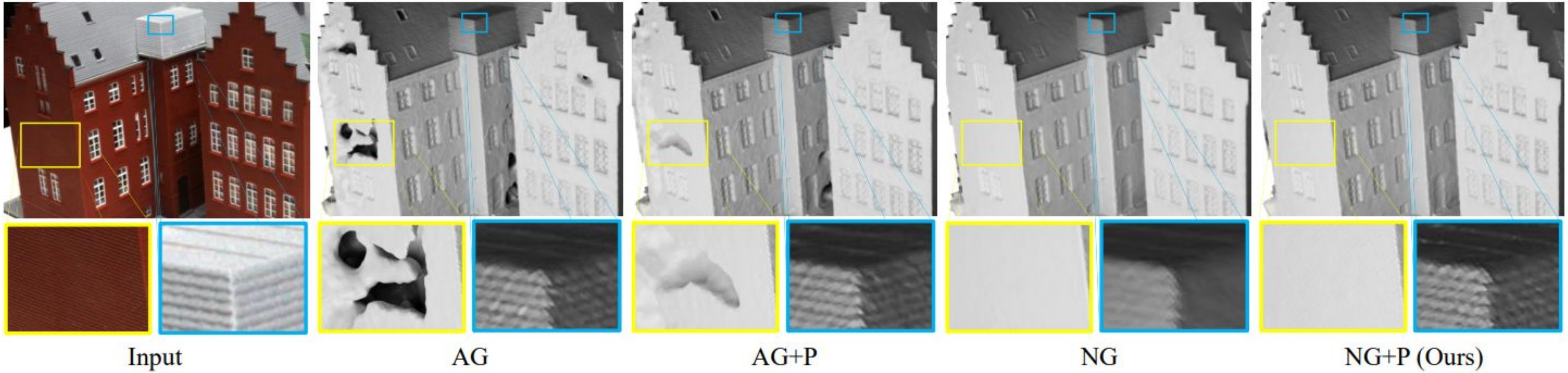
Small ϵ: Affects a smaller region, preserving finer details.



Progressive Levels of Details (Hash Grid Resolution)

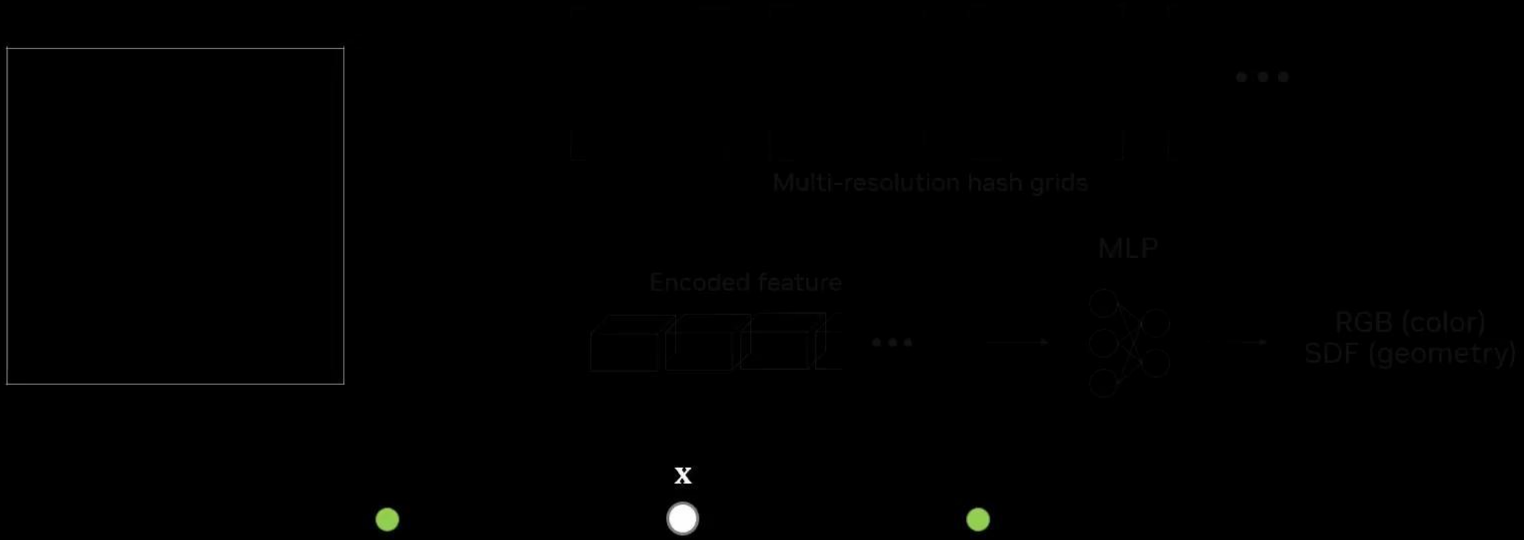
Begin with coarse hash grids.

Gradually enable finer grids to preserve and capture geometric details effectively.



Key ingredients of **Neuralangelo**:

2. Coarse-to-fine optimization for progressive level of details



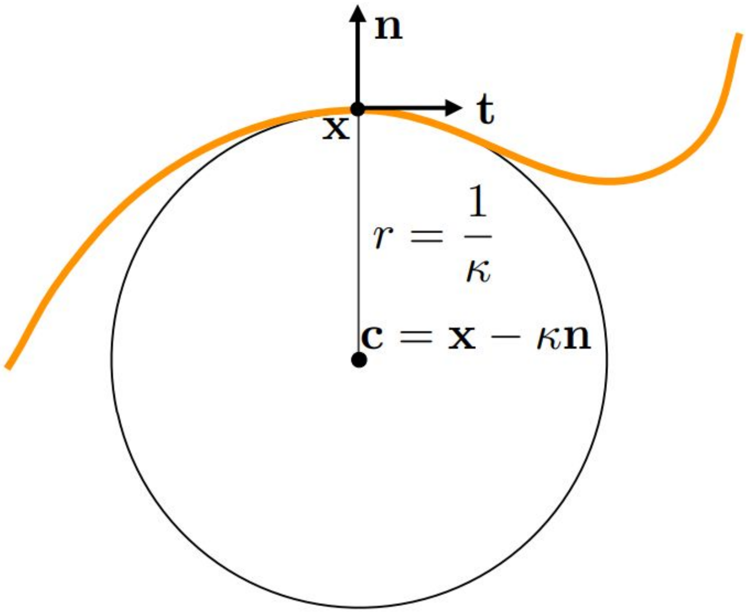
Curricular **encoding resolution** and

Curvature (2D)

Curvature κ: deviation from straight line

- Derivative of tangent:

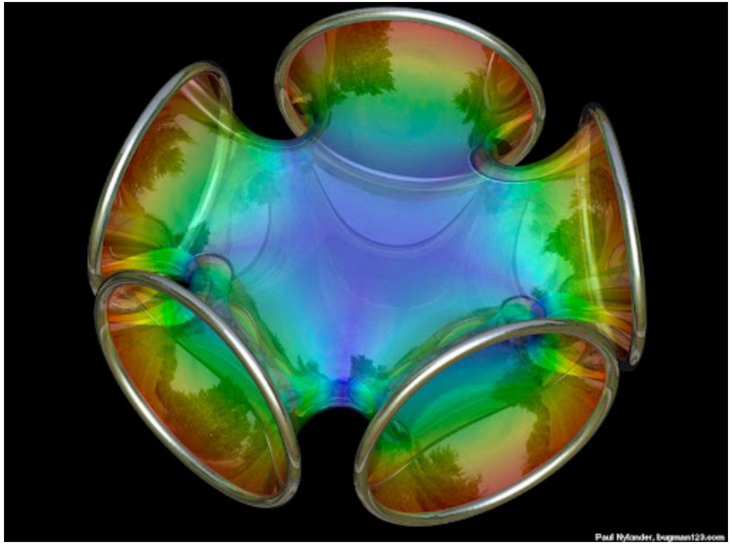
$$\mathbf{t}'(s) = \mathbf{x}''(s) =: \kappa\,\mathbf{n}(s) \longrightarrow \kappa := \|\mathbf{x}''(s)\|$$

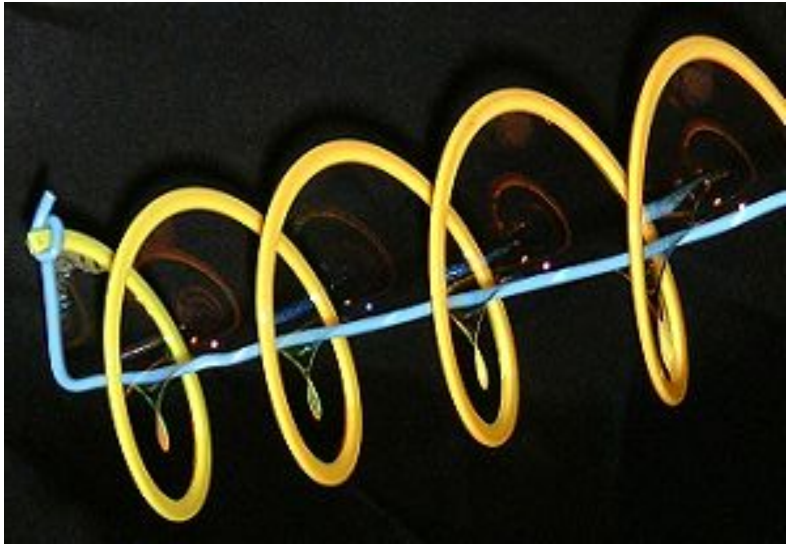


Minimal Surfaces

Surfaces minimizing mean curvature

- exact zero is possible.
- e.g. soap bubbles, elastics under tension.

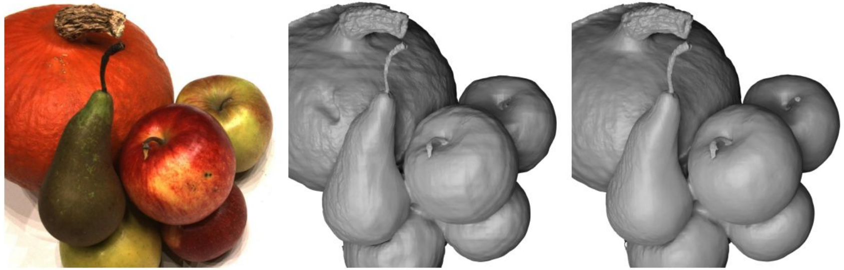




Regularizing the Mean Curvature

- Encourage the smoothness of reconstructed surfaces.

$$\mathcal{L}_{\text{curv}} = \frac{1}{N}\sum_{i=1}^{N}\left|\nabla^2 f(\mathbf{x}_i)\right|$$



(a) Input | w/o $\mathcal{L}_{\text{curv}}$ | w/ $\mathcal{L}_{\text{curv}}$

(b) Input | w/o topology warmup | w/ topology warmup

Optimization

$$\mathcal{L} = \mathcal{L}_{\text{RGB}} + w_{\text{eik}}\mathcal{L}_{\text{eik}} + w_{\text{curv}}\mathcal{L}_{\text{curv}}$$

RGB synthesis loss: RGB reconstruction loss between the input image and synthesized images.

Eikonal loss: regularize underlying SDF such that the surface normals are unit-norm.

Curvature loss: regularize underlying SDF such that the mean-curvature is not arbitrarily large.

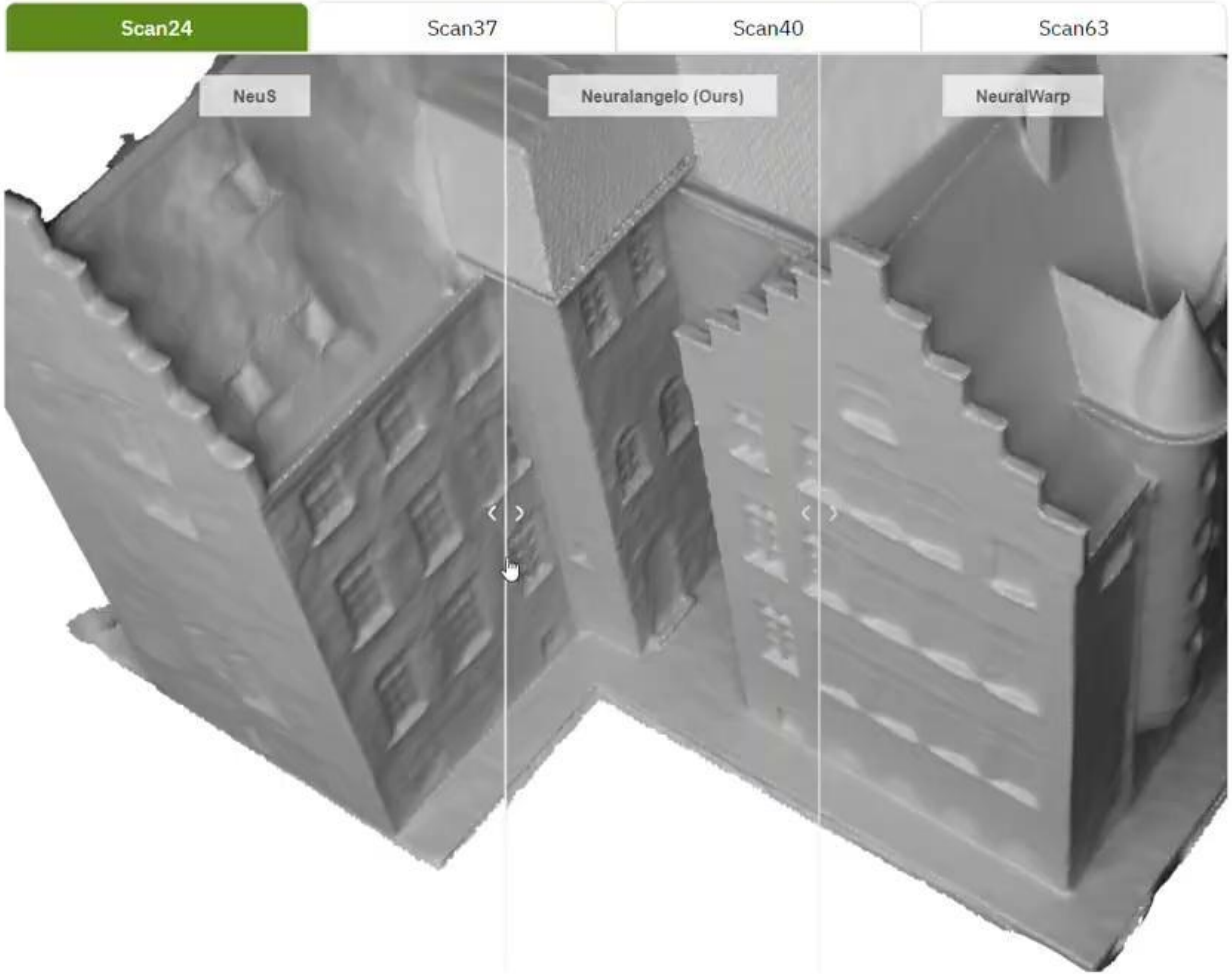
NVIDIA HQ Park

Quantitative Results

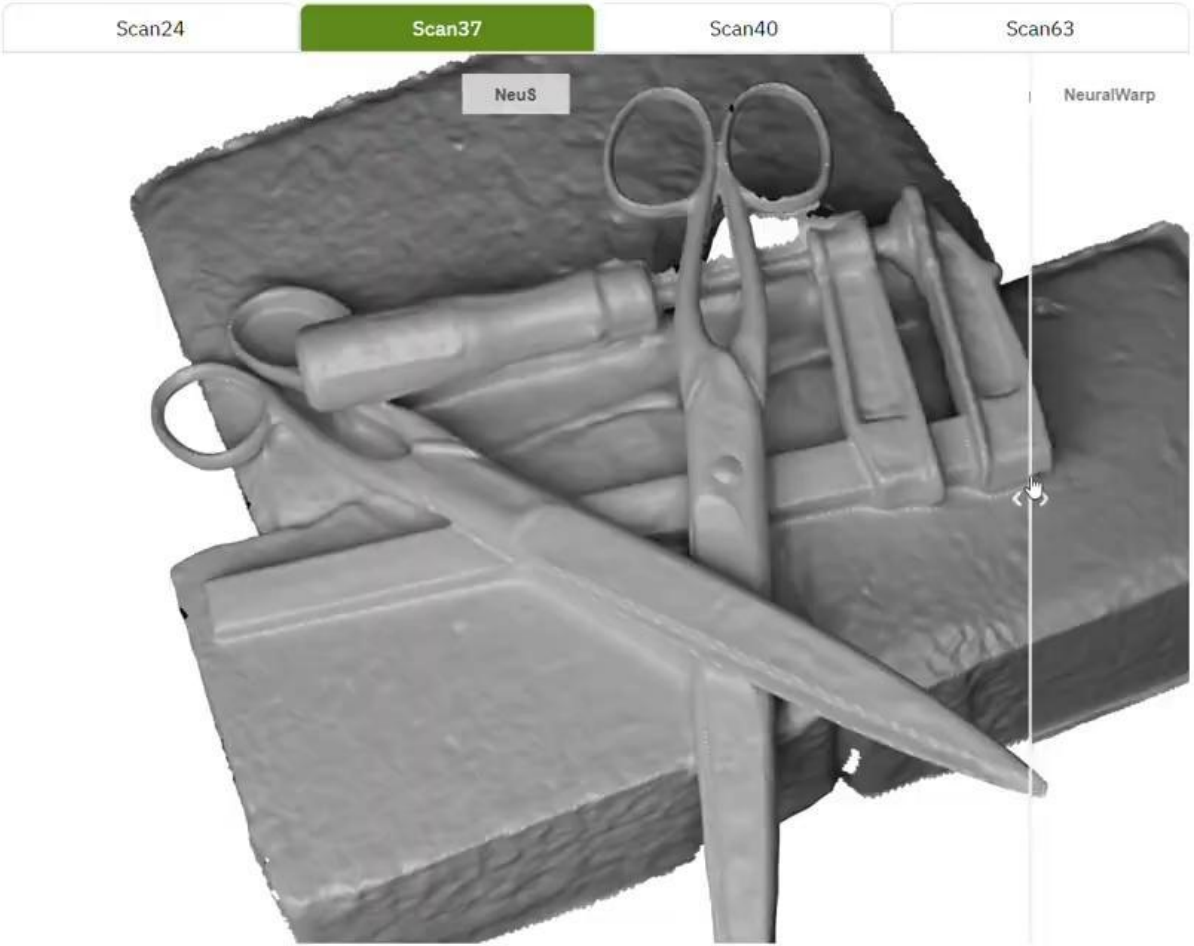
Metric	Method	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
Chamfer distance (mm) ↓	NeRF [25]	1.90	1.60	1.85	0.58	2.28	1.27	1.47	1.67	2.05	1.07	0.88	2.53	1.06	1.15	0.96	1.49
	VolSDF [47]	1.14	1.26	0.81	0.49	1.25	0.70	0.72	1.29	1.18	0.70	0.66	1.08	0.42	0.61	0.55	0.86
	NeuS [41]	1.00	1.37	0.93	0.43	1.10	0.65	**0.57**	1.48	1.09	0.83	0.52	1.20	0.35	0.49	0.54	0.84
	HF-NeuS [43]	0.76	1.32	0.70	0.39	1.06	0.63	0.63	**1.15**	1.12	0.80	0.52	1.22	0.33	0.49	0.50	0.77
	RegSDF [51] †	0.60	1.41	0.64	0.43	1.34	**0.62**	0.60	**0.90**	**0.92**	1.02	0.60	**0.59**	**0.30**	**0.41**	**0.39**	0.72
	NeuralWarp [3] †	0.49	**0.71**	**0.38**	0.38	**0.79**	0.81	0.82	1.20	1.06	**0.68**	0.66	**0.74**	0.41	0.63	0.51	0.68
	AG	0.67	1.04	0.84	0.39	1.43	1.23	1.11	1.24	1.54	0.85	0.50	1.01	0.37	0.51	0.44	0.88
	AG+P	0.59	0.95	0.46	**0.34**	1.19	0.70	0.79	1.19	1.37	**0.69**	**0.49**	0.93	0.33	**0.44**	0.44	0.73
	NG	**0.48**	0.81	0.43	**0.35**	0.89	0.71	0.61	1.26	1.06	0.74	**0.47**	0.79	0.33	0.45	**0.43**	**0.65**
	NG+P (Ours)	**0.37**	**0.72**	**0.35**	**0.35**	**0.87**	**0.54**	**0.53**	1.29	**0.97**	0.73	**0.47**	**0.74**	**0.32**	**0.41**	**0.43**	**0.61**
PSNR ↑	RegSDF [51] †	24.78	23.06	23.47	22.21	28.57	25.53	21.81	28.89	26.81	27.91	24.71	25.13	26.84	21.67	28.25	25.31
	NeuS [41]	26.62	23.64	26.43	25.59	30.61	**32.83**	29.24	33.71	26.85	31.97	32.18	28.92	28.41	35.00	34.81	29.79
	VolSDF [47]	26.28	25.61	26.55	26.76	31.57	31.50	29.38	33.23	28.03	32.13	33.16	31.49	30.33	34.90	34.75	30.38
	NeRF [25]	26.24	**25.74**	26.79	27.57	**31.96**	31.50	29.58	32.78	**28.35**	32.08	33.49	**31.54**	**31.00**	35.59	35.51	30.65
	AG	29.97	24.98	23.11	30.27	30.60	31.27	29.27	34.22	27.47	33.09	33.85	29.98	29.41	35.69	35.11	30.55
	AG+P	30.12	24.63	29.59	30.29	31.60	32.04	**29.85**	34.19	27.82	33.23	33.95	29.15	29.44	35.99	35.67	31.17
	NG	**30.34**	25.14	**30.20**	**30.79**	31.72	31.86	29.81	**34.36**	28.01	**33.45**	**34.38**	30.39	29.88	**36.02**	**35.74**	**31.47**
	NG+P (Ours)	**30.64**	**27.78**	**32.70**	**34.18**	**35.15**	**35.89**	**31.47**	**36.82**	**30.13**	**35.92**	**36.61**	**32.60**	**31.20**	**38.41**	**38.05**	**33.84**

Quantitative results on DTU dataset

Object-centric Reconstruction

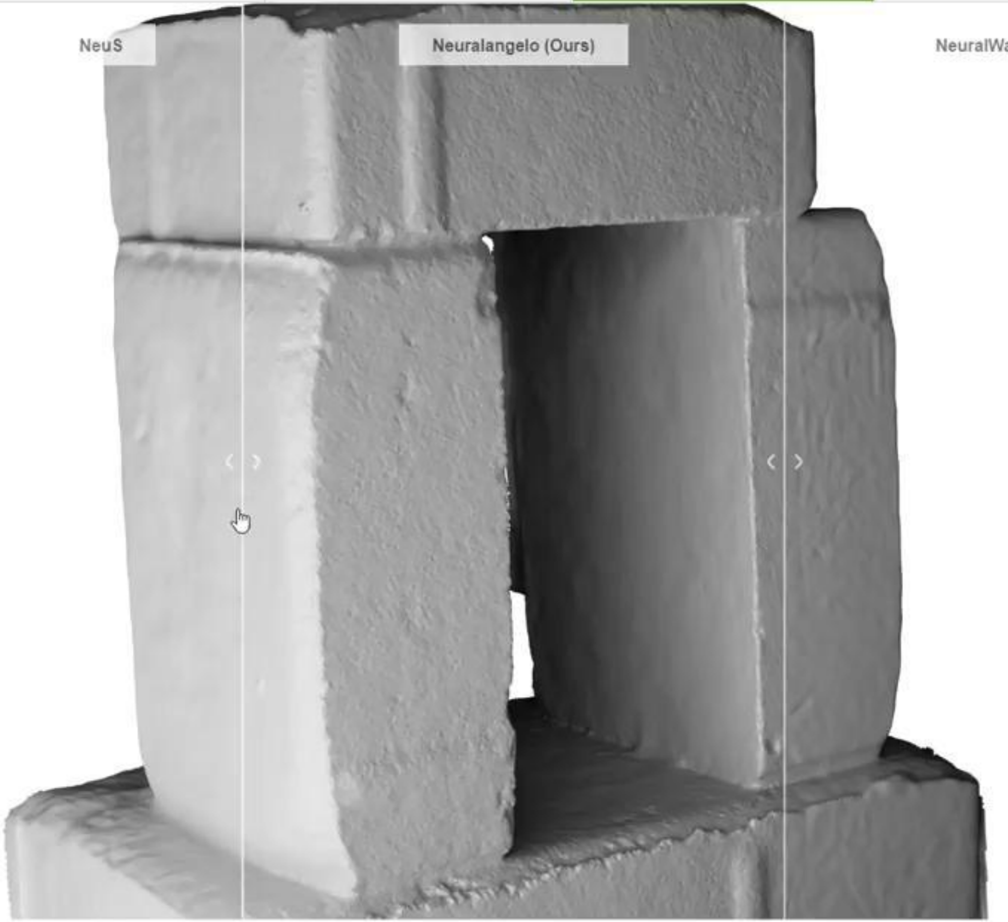


Object-centric Reconstruction



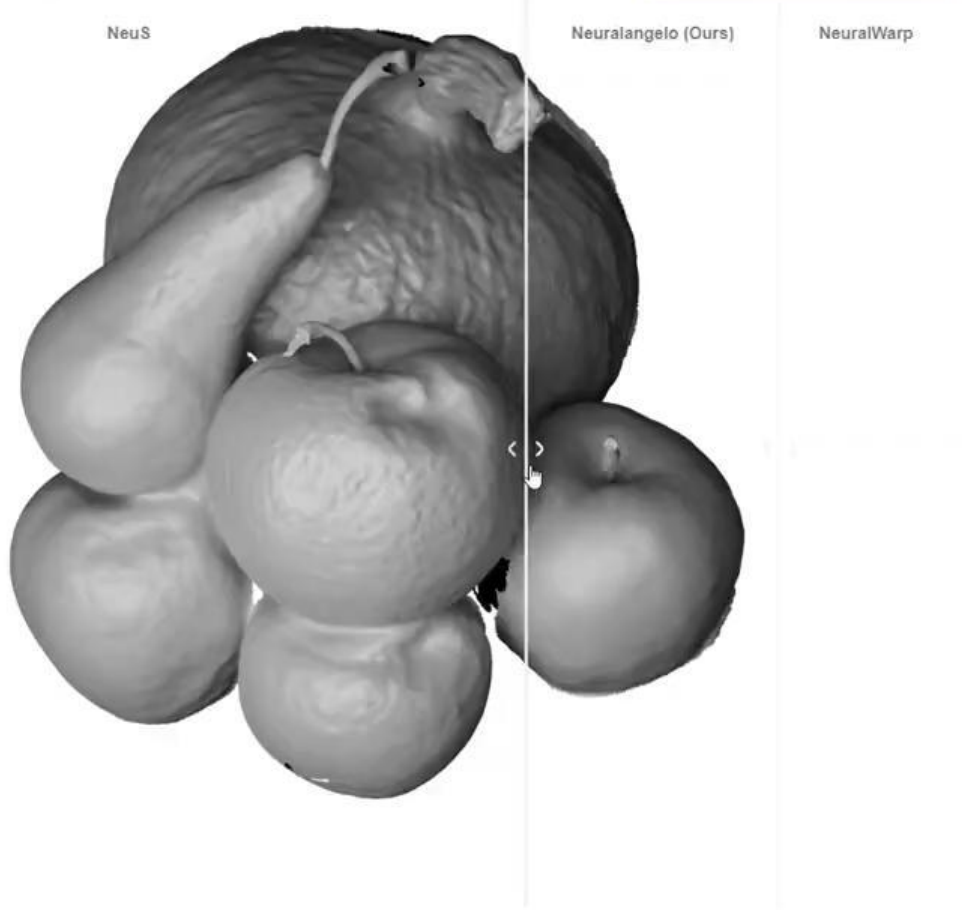
Object-centric Reconstruction

Scan24	Scan37	**Scan40**	Scan63



Object-centric Reconstruction

Scan24	Scan37	Scan40	Scan63



Thank you!

- Thank you for your attention!
- I appreciate your time and interest.
- If you have any questions, please feel free to ask.
- Contact information: alimohammadiamirhossein@gmail.com