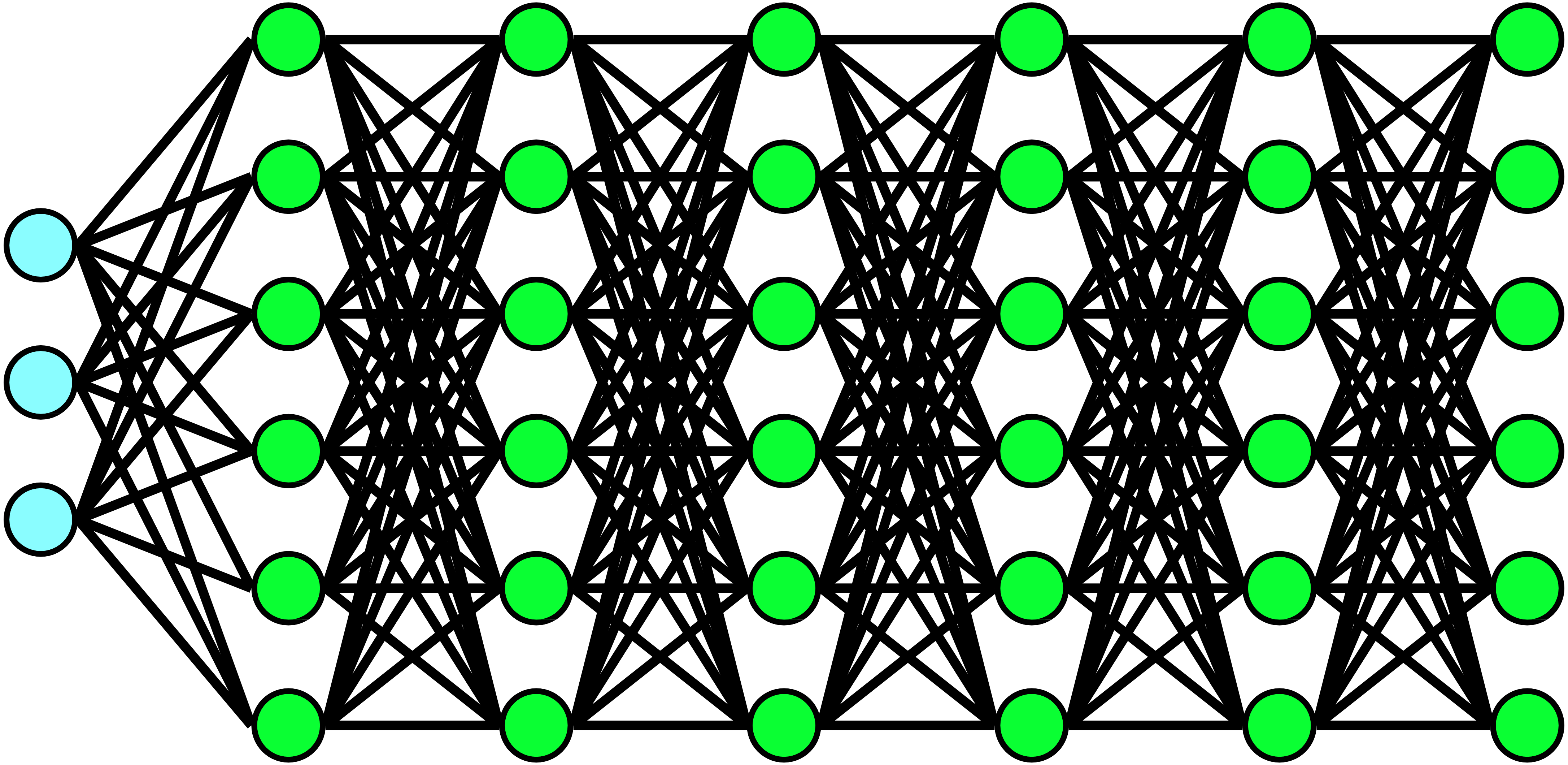
Overview

0. Fundamentals of Classical Rendering Techniques in Computer Graphics

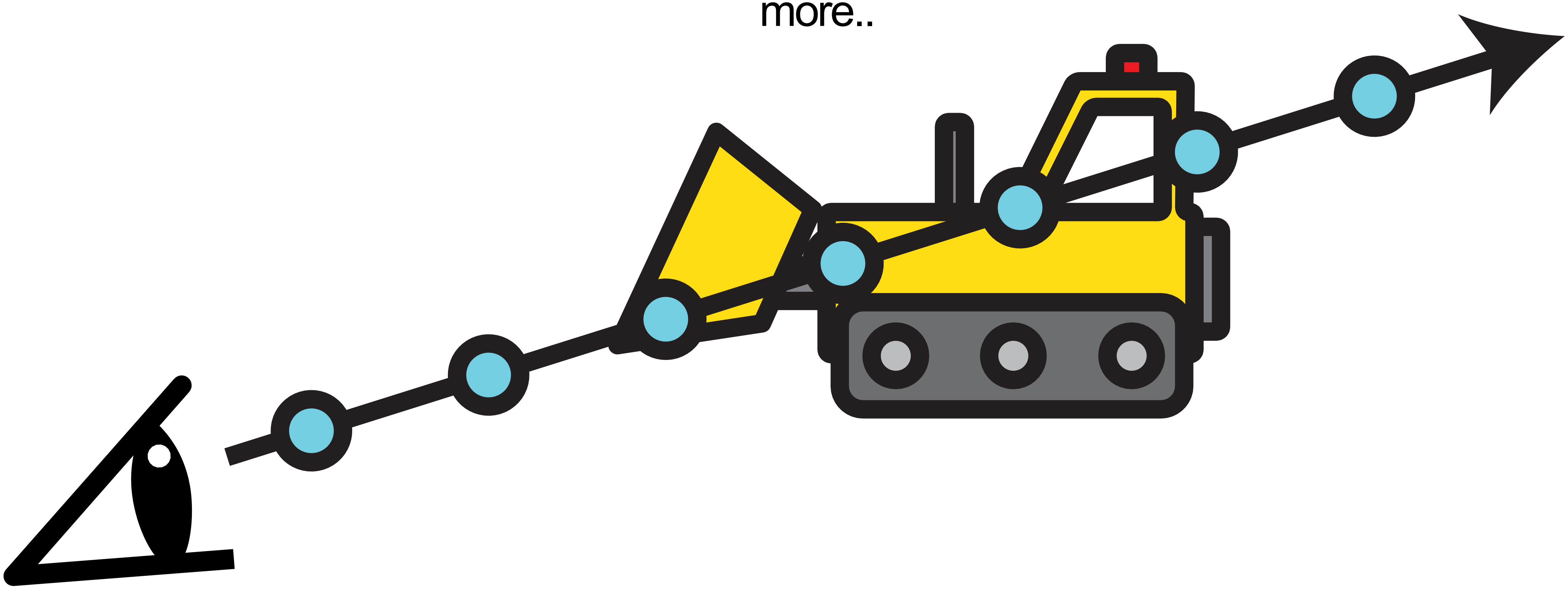
1. Three pioneering works in Neural Scene Representations and Neural Rendering
- Scene Representation Networks (SRN)
- Neural Volumes (before that: Deep Appearance Models)
- Neural Radiance Fields (NeRF)

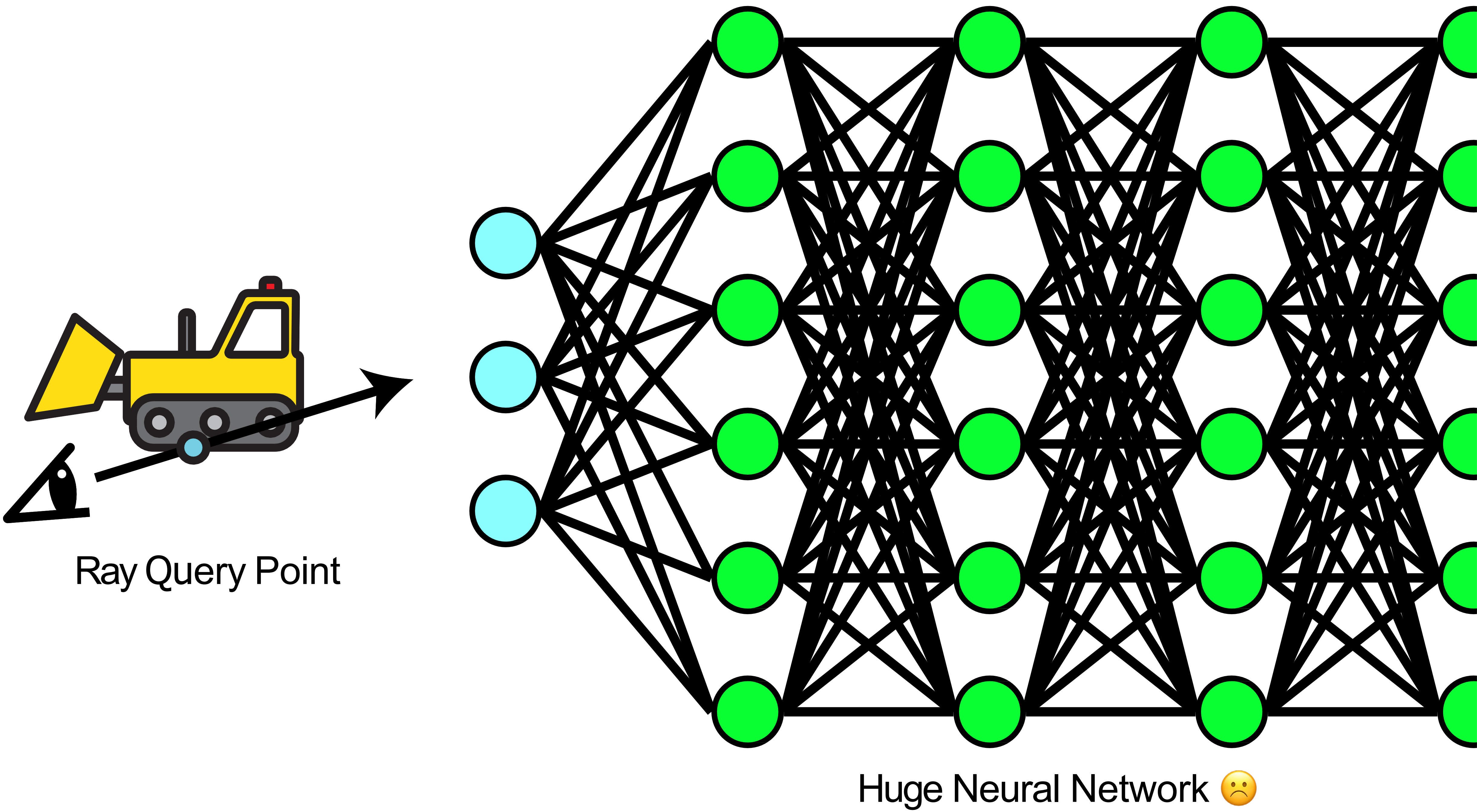
2. Different Neural Scene Representations
- Uniform Grids -> Sparse Grids -> Multiresolution Grids -> Hash Grids
- Point Clouds / Gaussian Splats
- Surface Mesh / Volumetric Mesh (Tetrahedron)
- Multiplane Images

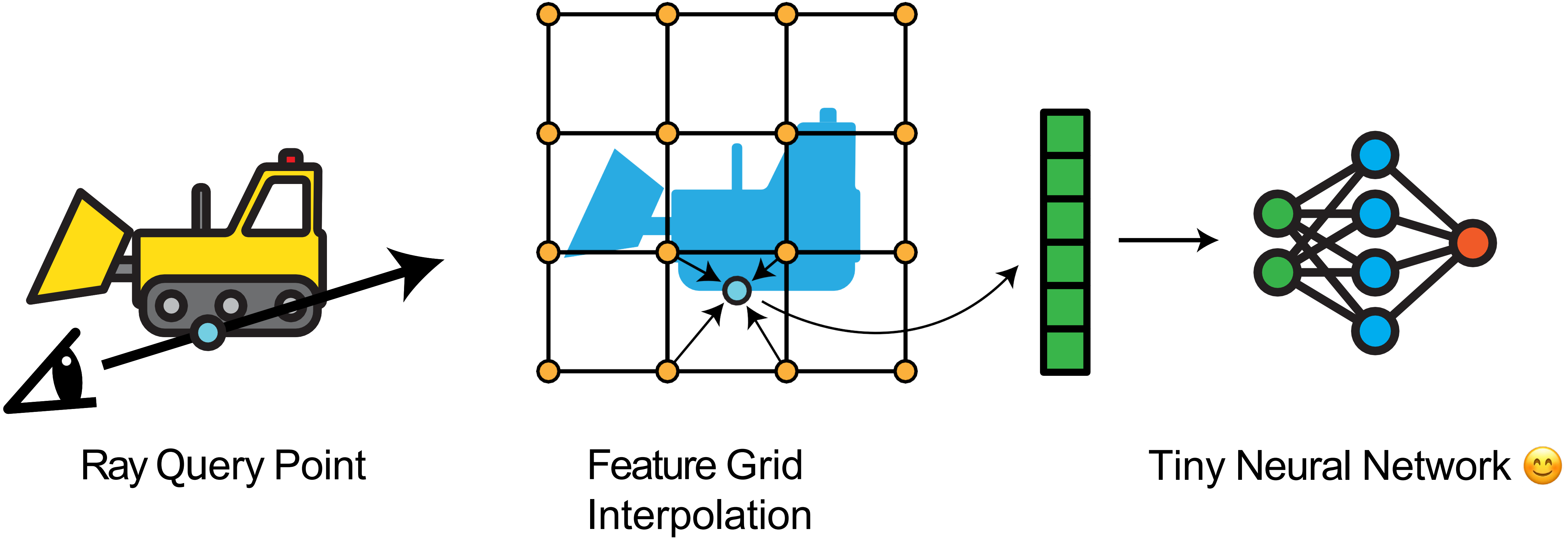
Hybrid Representations



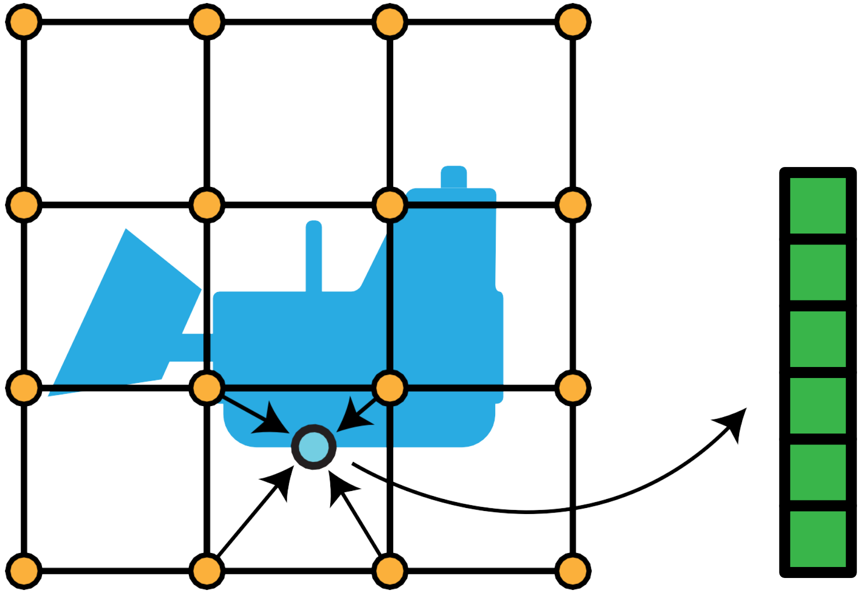
Usually 128, 256, 512 samples or more..




Ray Query Point
Huge Neural Network ☹

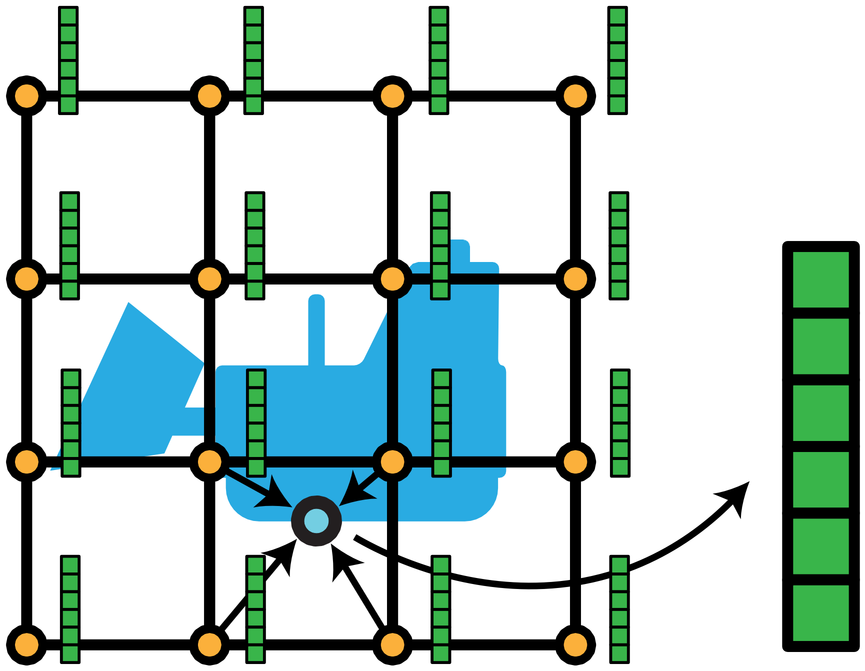

Ray Query Point
Feature Grid Interpolation
Tiny Neural Network 😊

Hybrid Representations



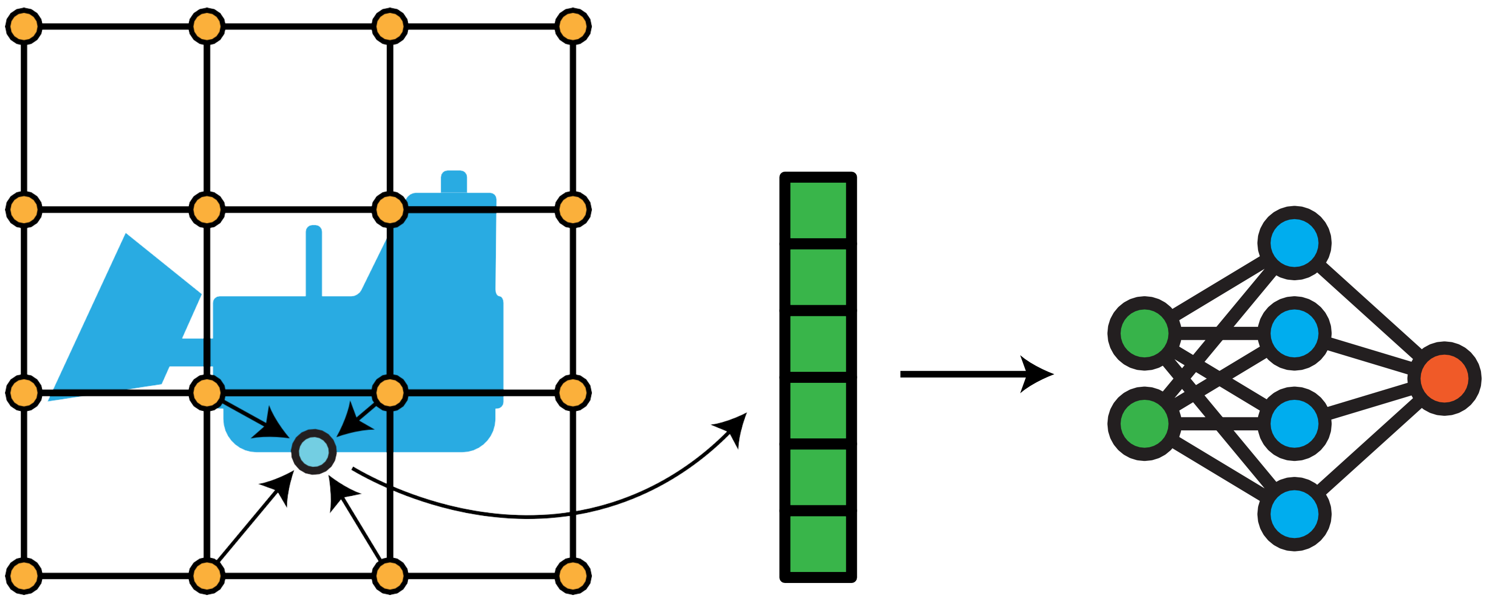
Feature Grid
Interpolation

Hybrid Representations



Feature Grid
Interpolation

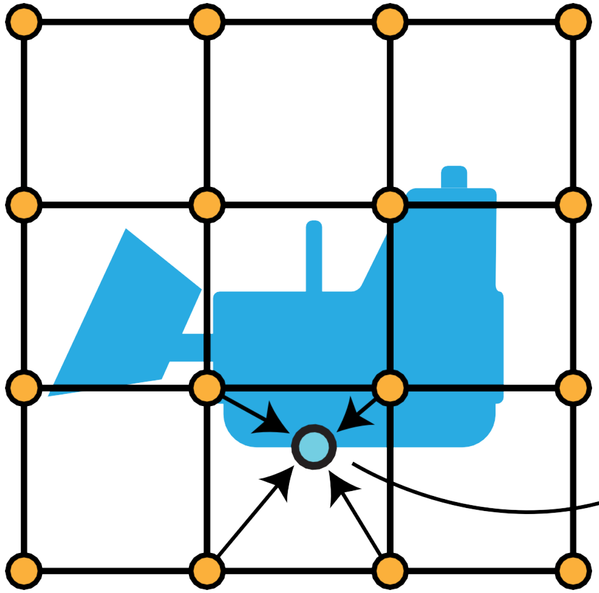
Hybrid Representations



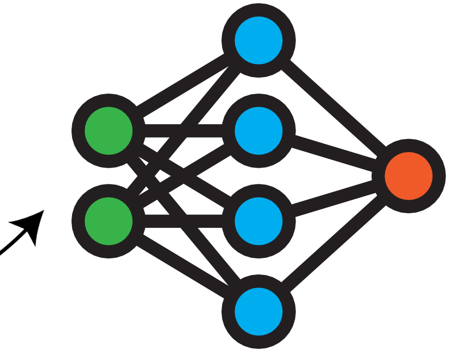
Feature Grid
Interpolation

Tiny Neural Network 😊

Hybrid Representations



Feature Grid
Interpolation



Tiny Neural Network 😊

Hybrid Representations

Hybrid Representations for Neural Fields in 2013!

Global Illumination with Radiance Regression Functions

Peiran Ren*‡ Jiaping Wang† Minmin Gong‡ Stephen Lin‡ Xin Tong‡ Baining Guo‡*
*Tsinghua University †Microsoft Corporation ‡Microsoft Research Asia

Figure 1: *Real-time rendering results with radiance regression functions for scenes with glossy interreflections (a), multiple local lights (b), and complex geometry and materials (c).*

Abstract

We present radiance regression functions for fast rendering of global illumination in scenes with dynamic local light sources. A radiance regression function (RRF) represents a non-linear mapping from local and contextual attributes of surface points, such as position, viewing direction, and lighting condition, to their indirect illumination values. The RRF is obtained from precomputed shading samples through regression analysis, which determines a function that best fits the shading data. For a given scene, the shading samples are precomputed by an offline renderer.

The key idea behind our approach is to exploit the nonlinear coherence of the indirect illumination data to make the RRF both compact and fast to evaluate. We model the RRF as a multilayer acyclic feed-forward neural network, which provides a close functional approximation of the indirect illumination and can be efficiently evaluated at run time. To effectively model scenes with spa-

Keywords: global illumination, real time rendering, neural network, non-linear regression

1 Introduction

Global light transport provides scenes with visually rich shading effects that are an essential component of photorealistic rendering. Much of the shading detail arises from multiple bounces of light. This reflected light, known as indirect illumination, is generally expensive to compute. The most successful existing approach for indirect illumination is precomputed radiance transfer (PRT) [Sloan et al. 2002; Ramamoorthi 2009], which precomputes the global light transport and stores the resulting PRT data for fast rendering at run time. However, even with PRT, real-time rendering with dynamic viewpoint and lighting remains difficult.

Two major challenges in real-time rendering of indirect illumina-

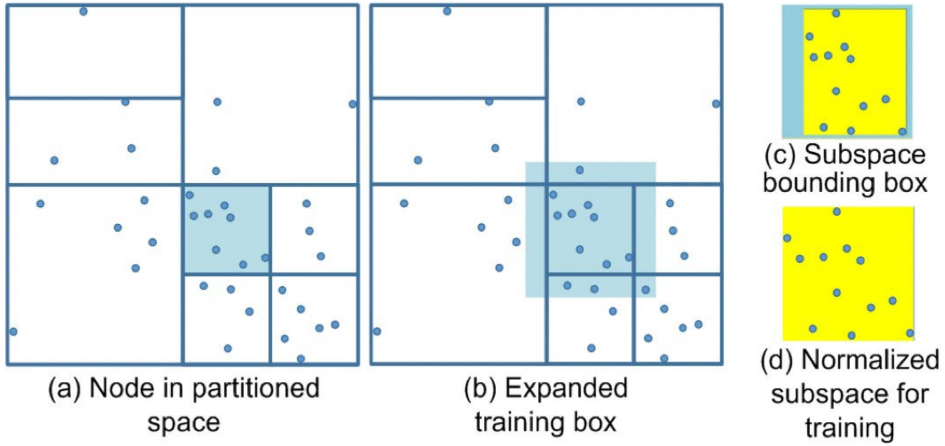
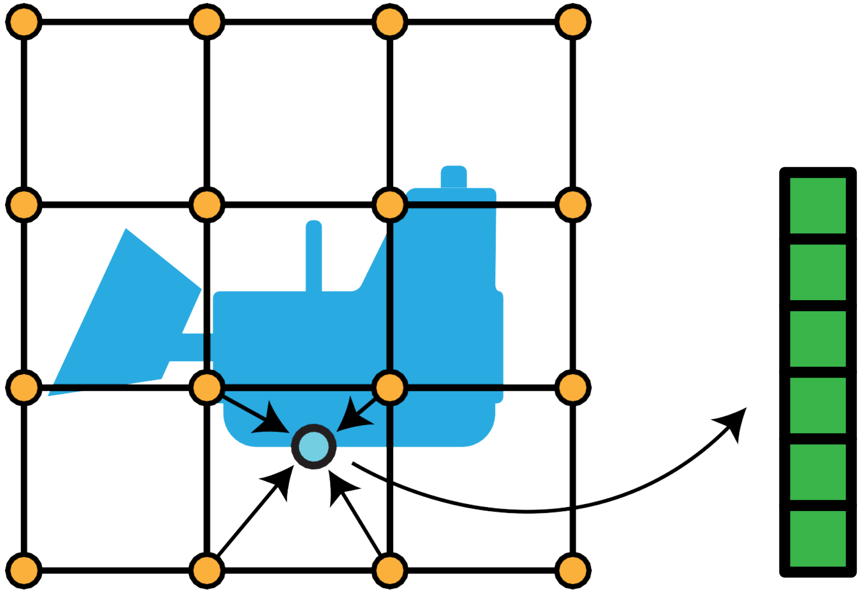


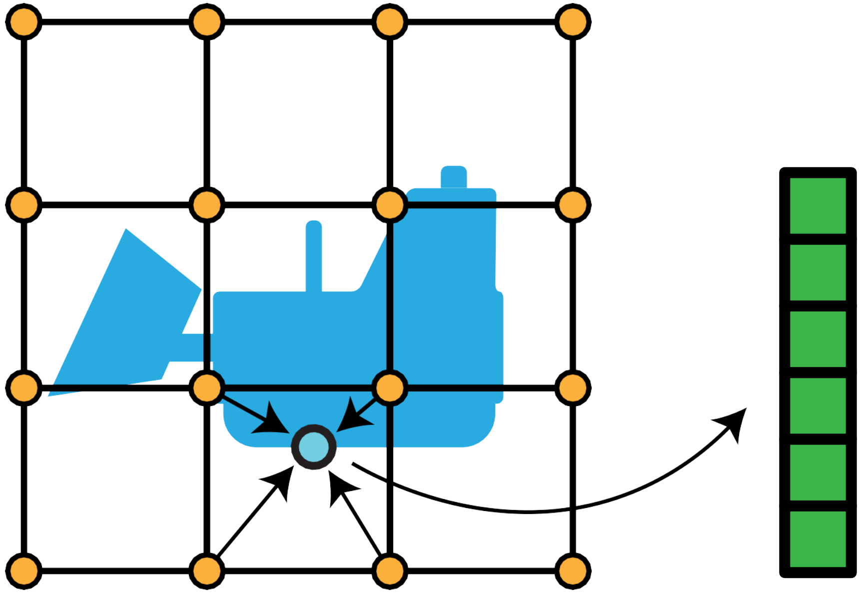
Figure 4: *Partitioning of input space for fitting of multiple RRFs.*

Uniform Grids



[PIFu (Saito et al.), Neural Volumes (Lombardi et al.), etc]

Uniform Grids



[PIFu (Saito et al.), Neural Volumes (Lombardi et al.), etc]

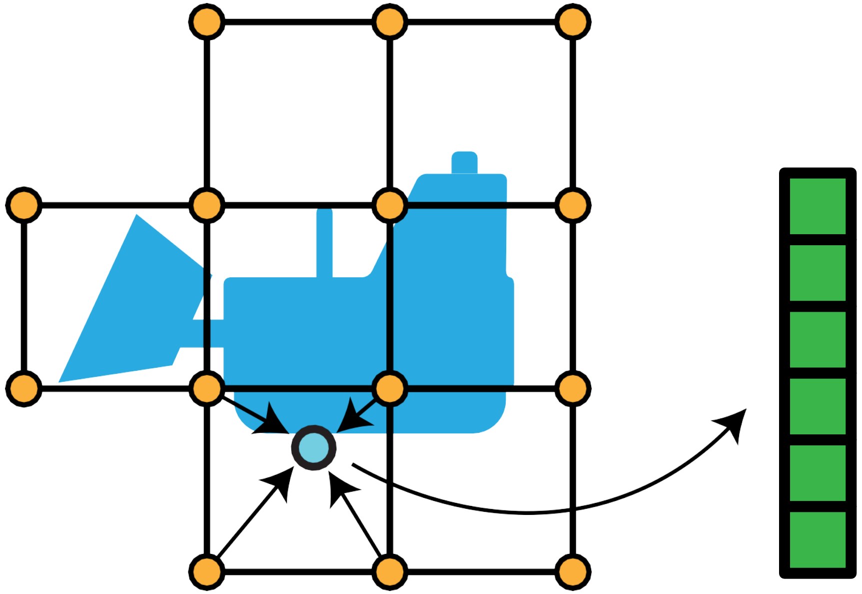
Pros:

- Easy to implement
- Algorithmically fast access
- Established operations like convolutions
- Simple topology

Cons:

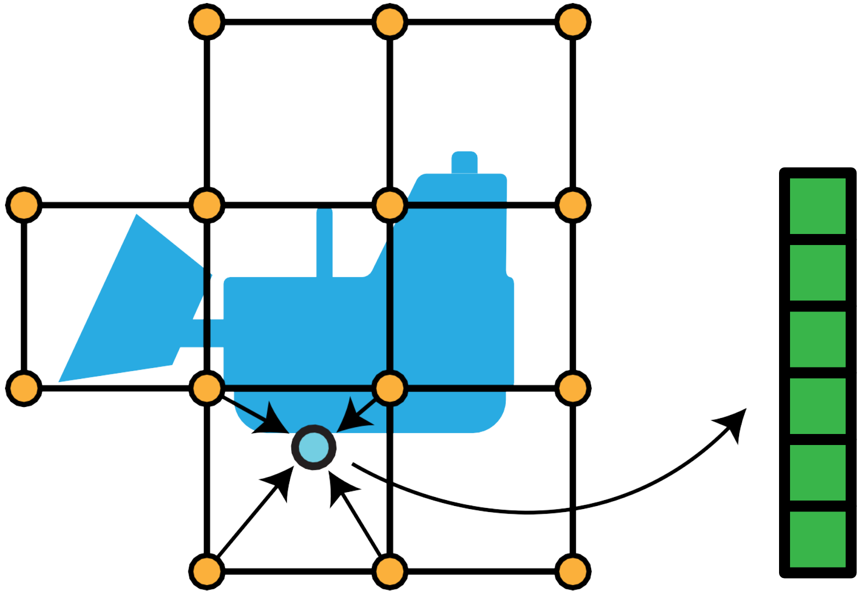
- Expensive in memory and bandwidth

Sparse Grids



[DeepLS (Chabra et al.), NSVF (Liu et al.), NGLOD (Takikawa et al.), etc]

Sparse Grids



[DeepLS (Chabra et al.), NSVF (Liu et al.), NGLOD (Takikawa et al.), etc]

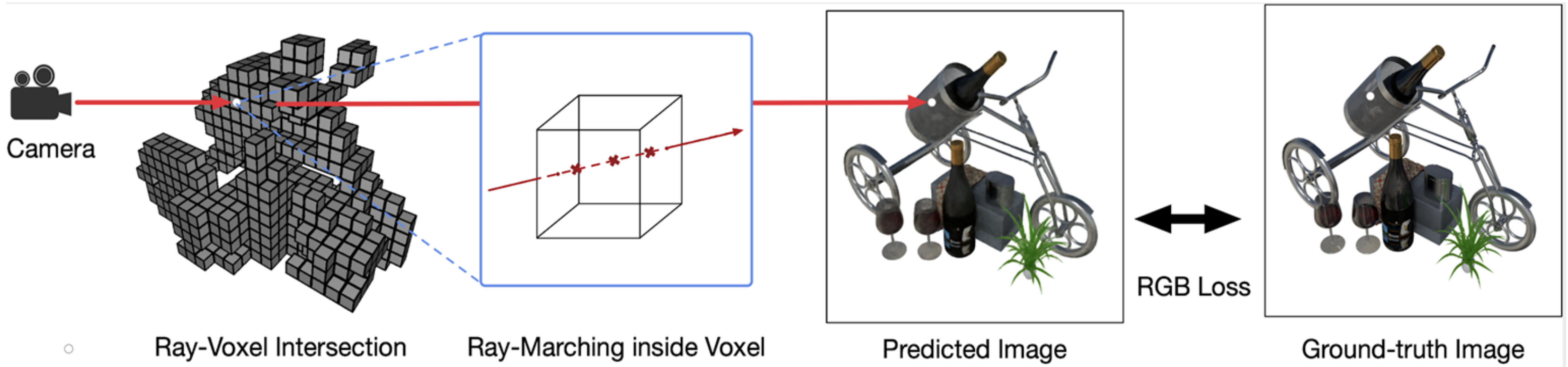
Pros:

- Memory Efficient
- Algorithmically efficient access
- GPU-compatible data structures
- Established operations like sparse 3D convs

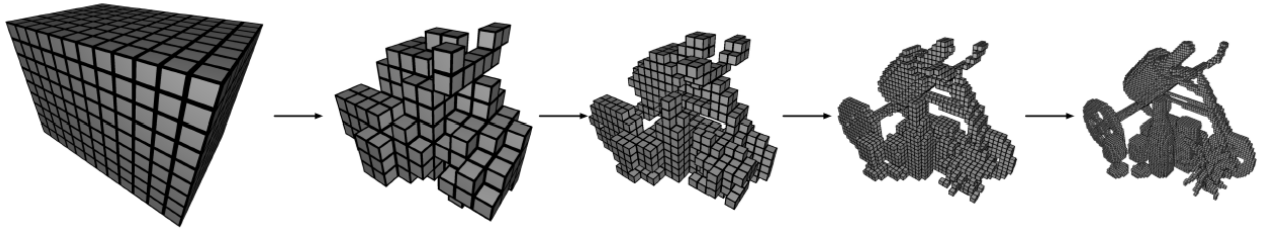
Cons:
- Need to manage a complex data structure
- Topology hard to generate

NSVF (Liu et al)

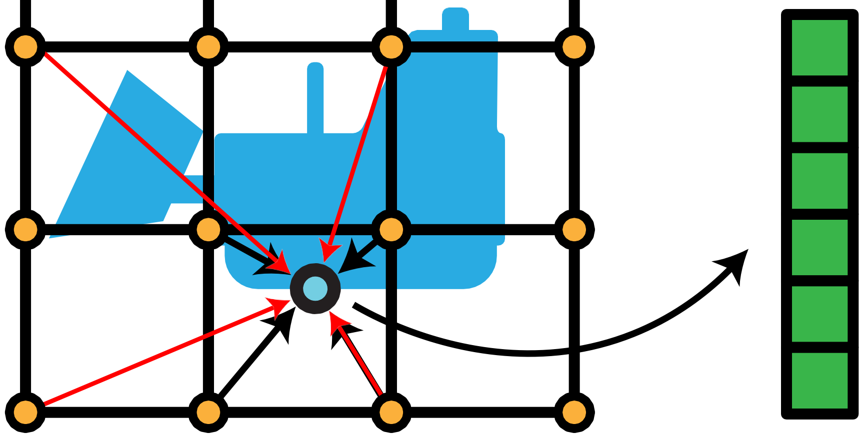
Scene Representation - Neural Sparse Voxel Fields (NSVF).
Volume Rendering with NSVF.



Progressive Learning: we train NSVF progressively with the differentiable volume rendering operation from a set of posed 2D images.

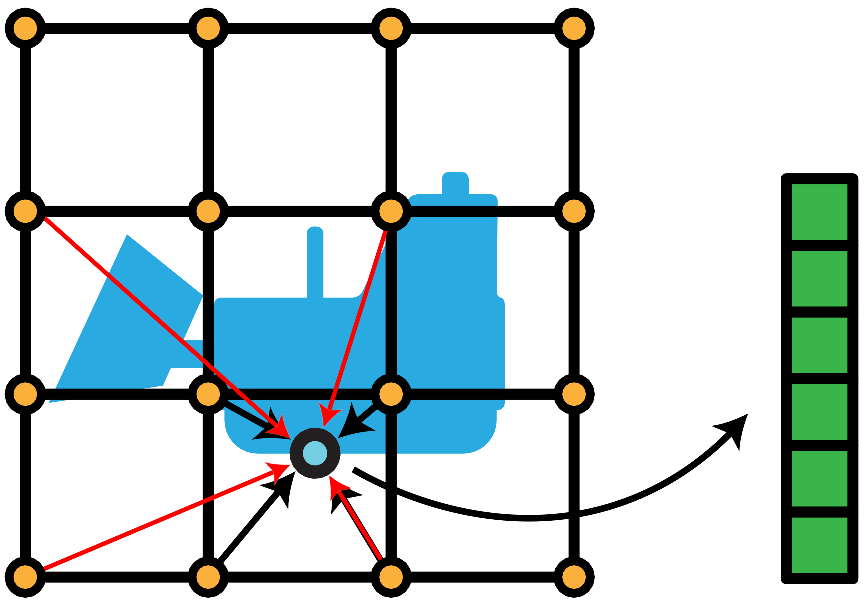


Multiresolution Grids



[NGLOD (Takikawa et al.), ACORN (Lindell et al.), Instant-NGP (Muller et al.), etc]

Multiresolution Grids



[NGLOD (Takikawa et al.), ACORN (Lindell et al.), Instant-NGP (Muller et al.), etc]

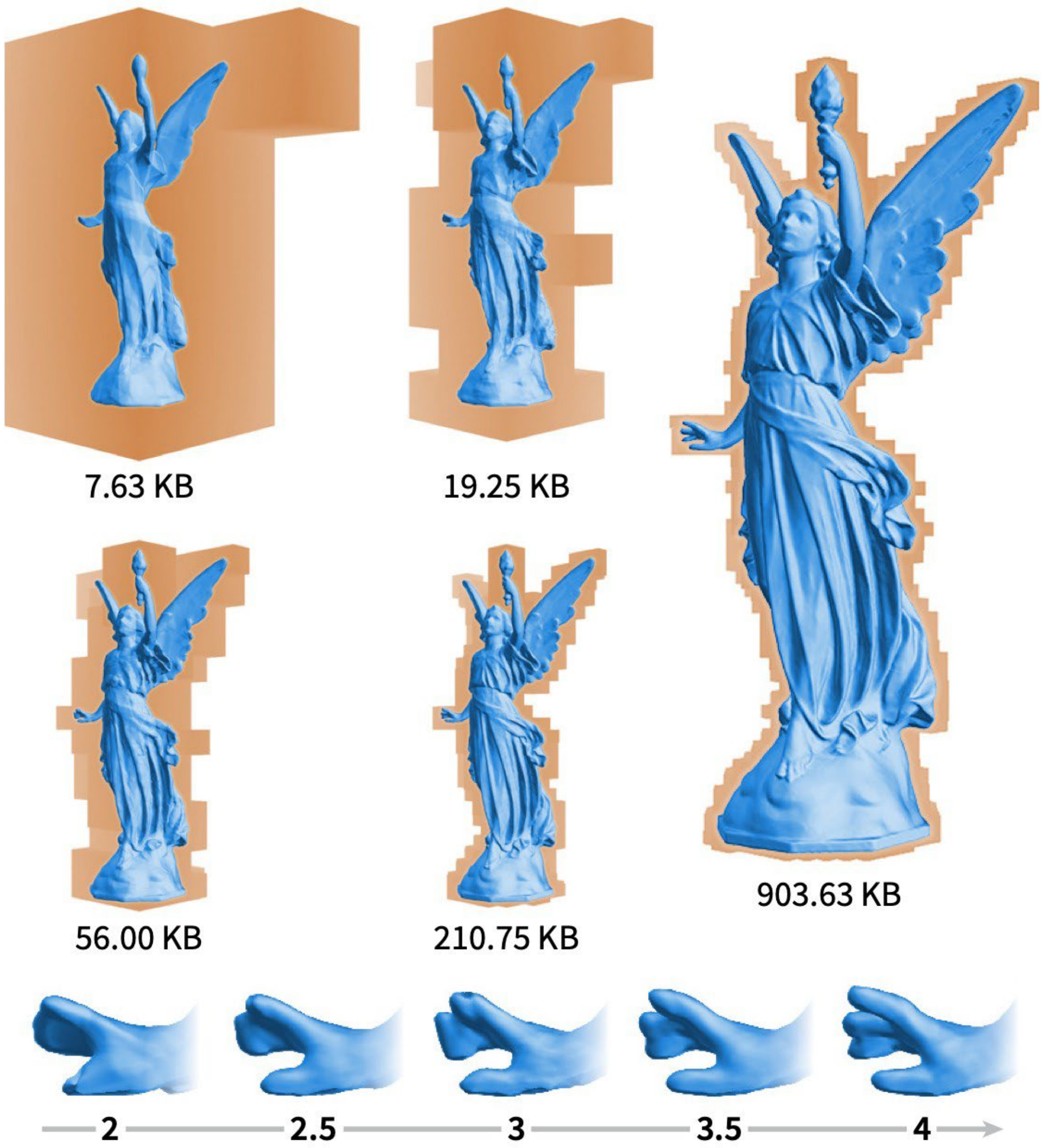
Pros:

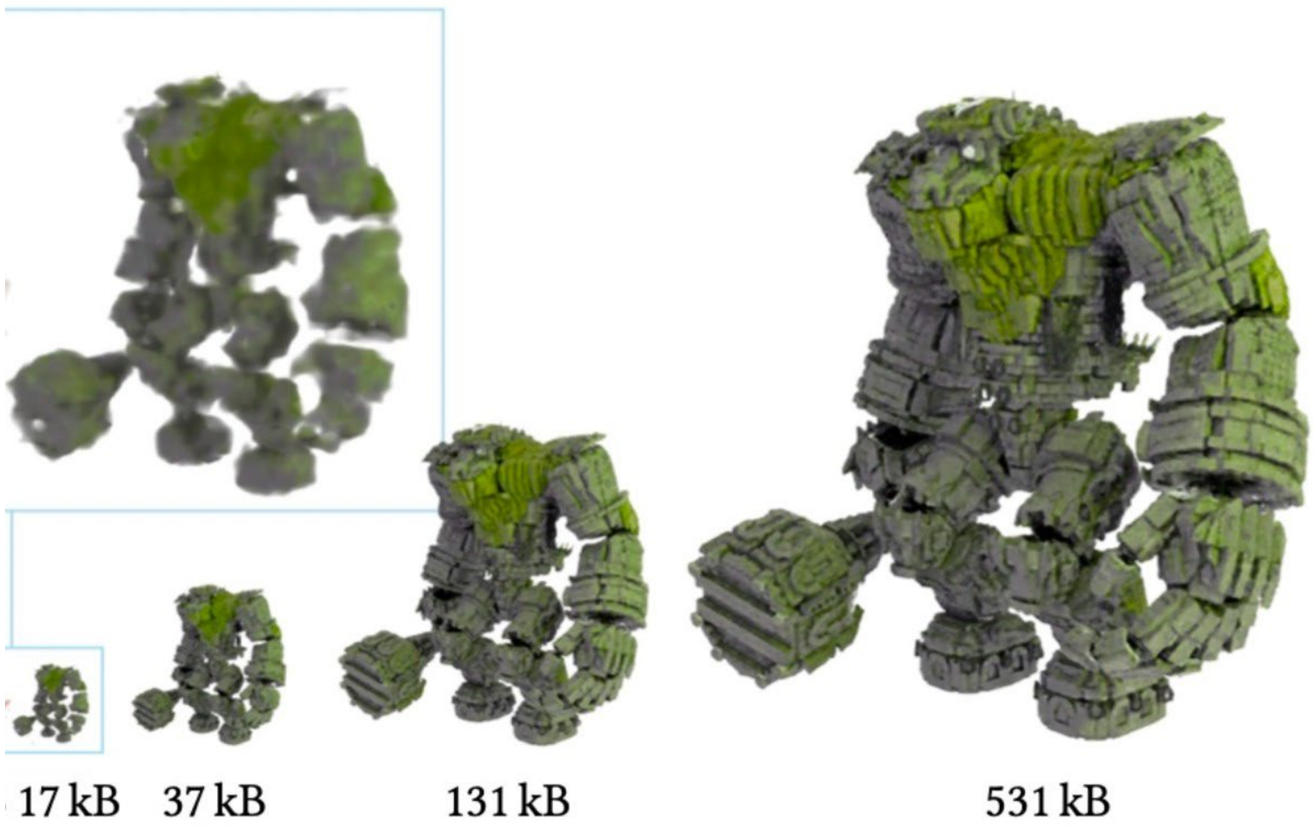
- Multiple streaming levels of detail (LOD)
- Wider support region

Cons:

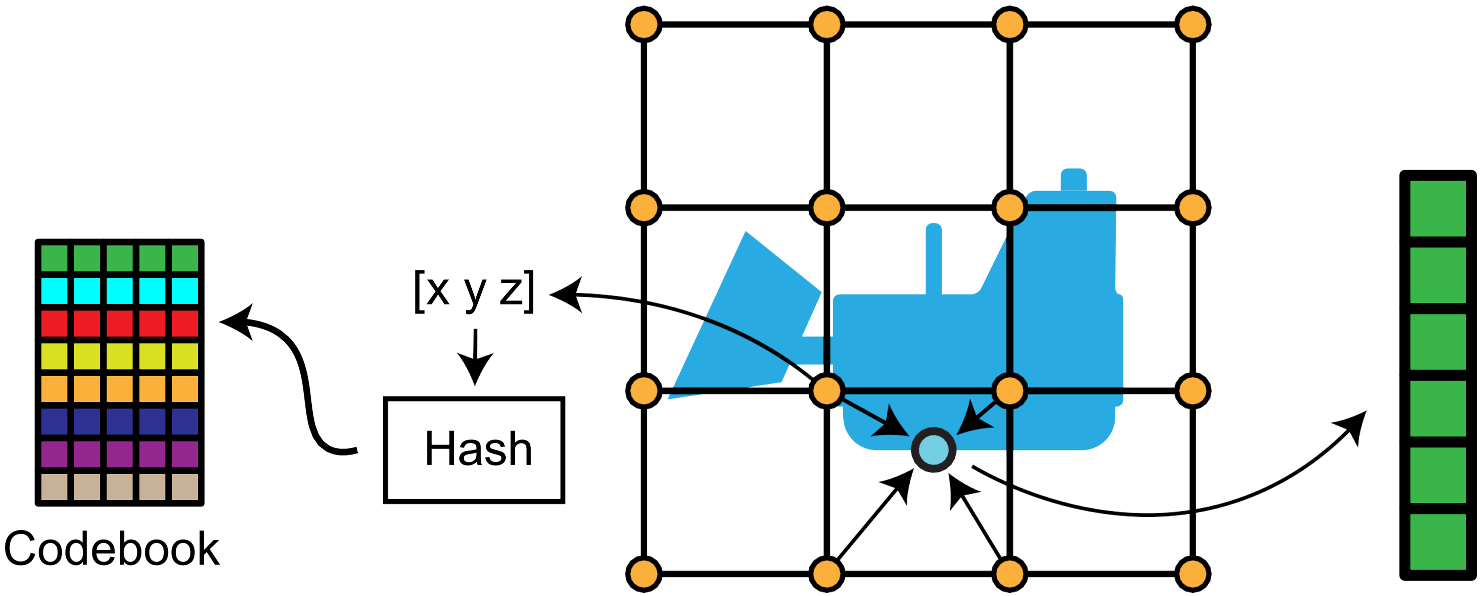
- More memory
- More complexity

NGLOD (Takikawa et al.)



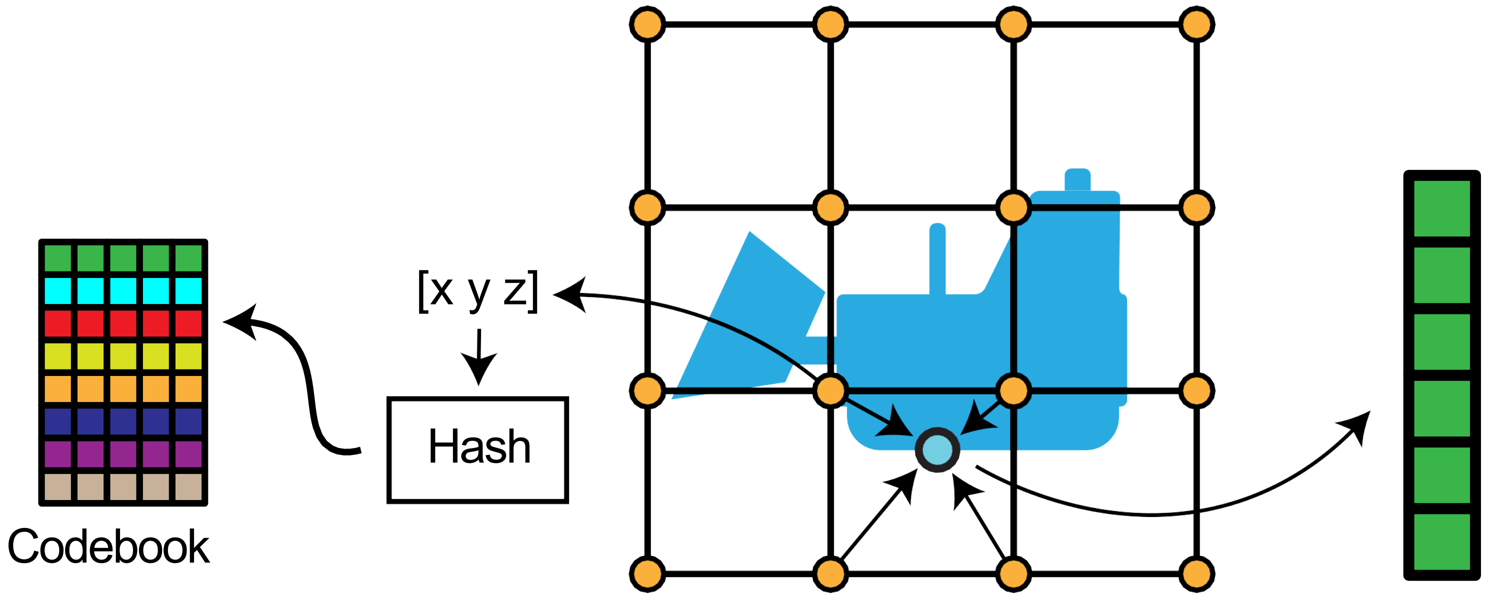


Hash Grids



[Instant-NGP (Muller et al.)]

Hash Grids



[Instant-NGP (Muller et al.)]

Pros:

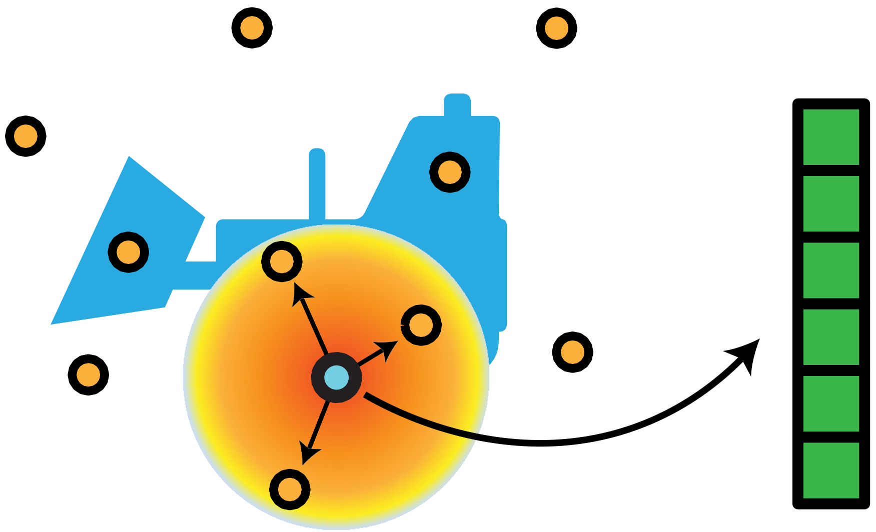
- Disaggregate resolution from memory cost
- No complex data structures
- Performant memory access if codebook is small enough

Cons:

- Multiresolution and large codebooks needed for collision resolution
- Features not spatially local

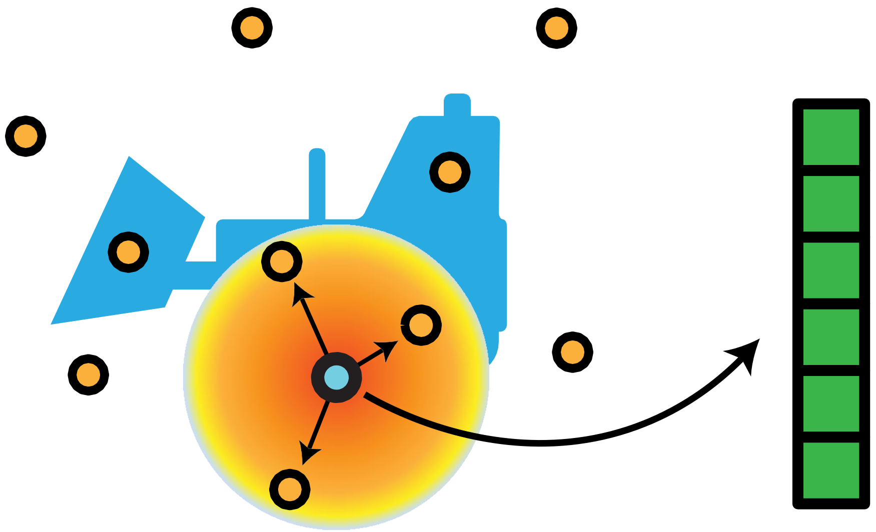
Instant-NGP (Müller et al.)

Point Clouds



[Liu et al. 2019, LDIF (Genova et al.), 3DILG (Zhang et al.) etc]

Point Clouds



[Liu et al. 2019, LDIF (Genova et al.), 3DILG (Zhang et al.) etc]

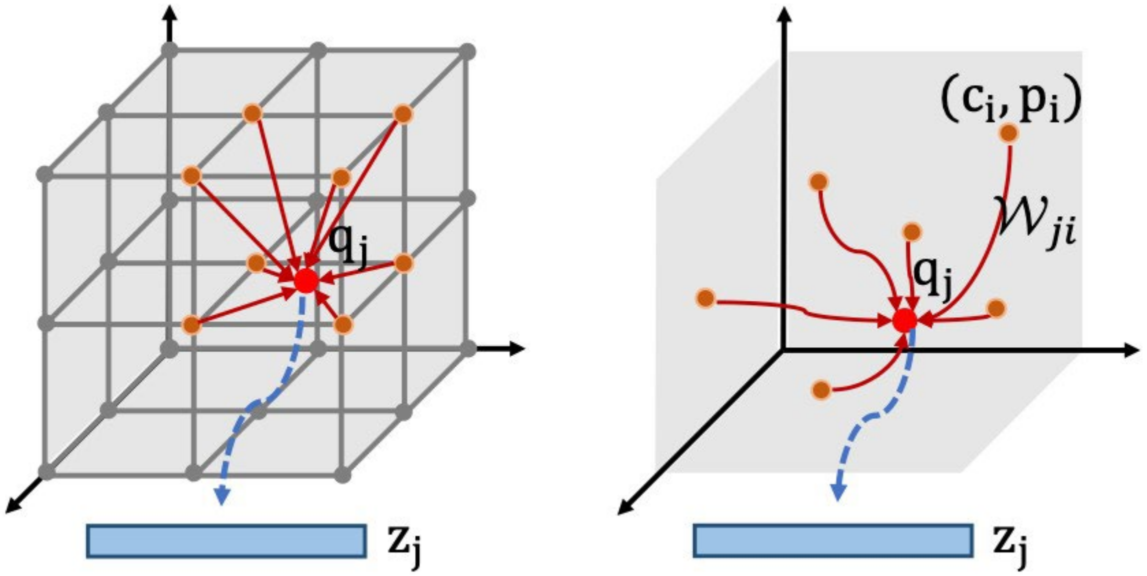
Pros:

- Can be densely supported in space
- Expressive

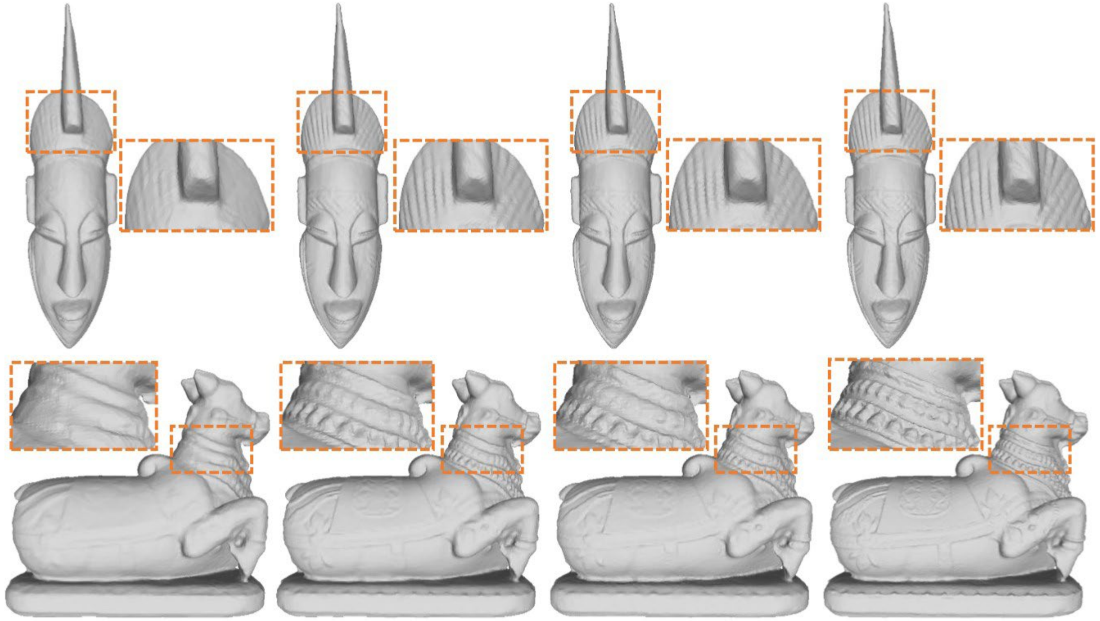
Cons:

- Often needs complex data structures for fast access and interpolation

DCC-DIF (Li et al.)



(a) Interpolation of latent grids (b) Interpolation of ours

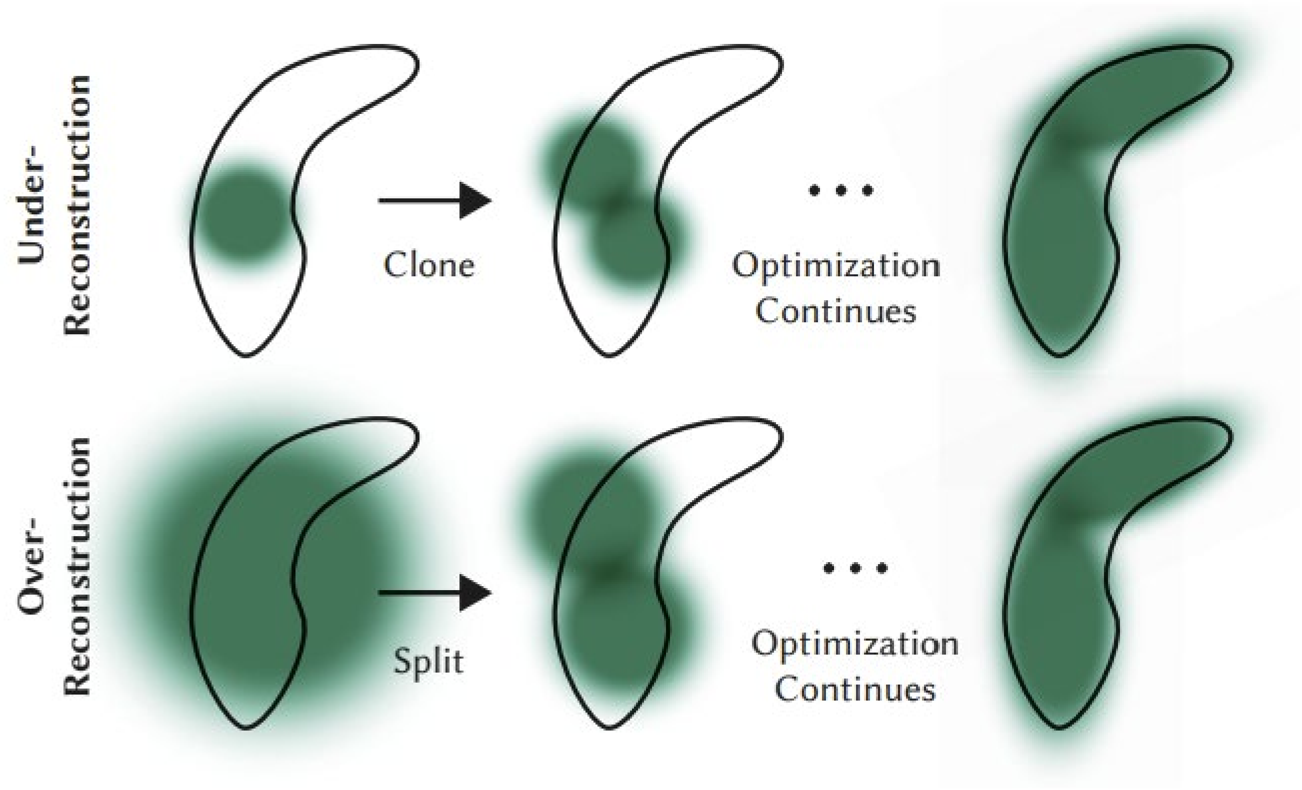


(a) NGLOD3 (b) NGLOD5 (c) Ours (d) Reference

NGLOD3 [38]	NGLOD4 [38]	NGLOD5 [38]	Ours
99.0	99.3	99.4	**99.5**
3.69	3.59	3.57	**3.55**
5.7K/0.9K	41.7K/3.7K	316K/15K	5.6K
4.7K	4.7K	4.7K	4.7K

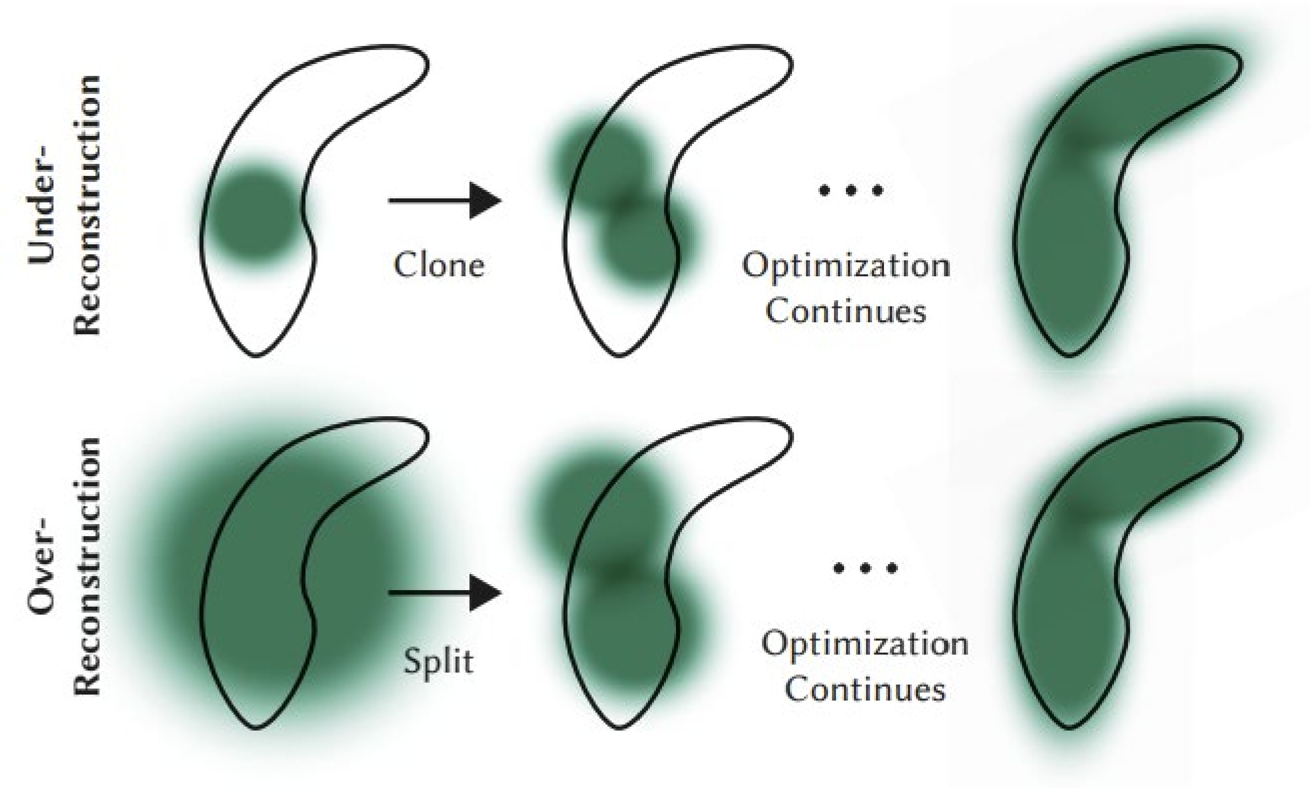
[Li et al.]

Gaussian Splats



[Kerbl et al. 2019]

Gaussian Splats



[Kerbl et al. 2019]

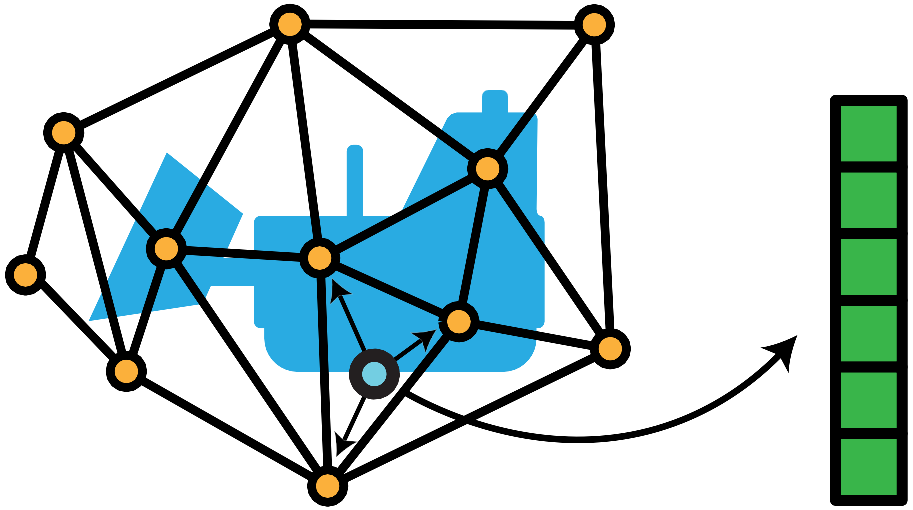
Pros:

- Expressive
- Compatible with efficient rasterization

Cons:

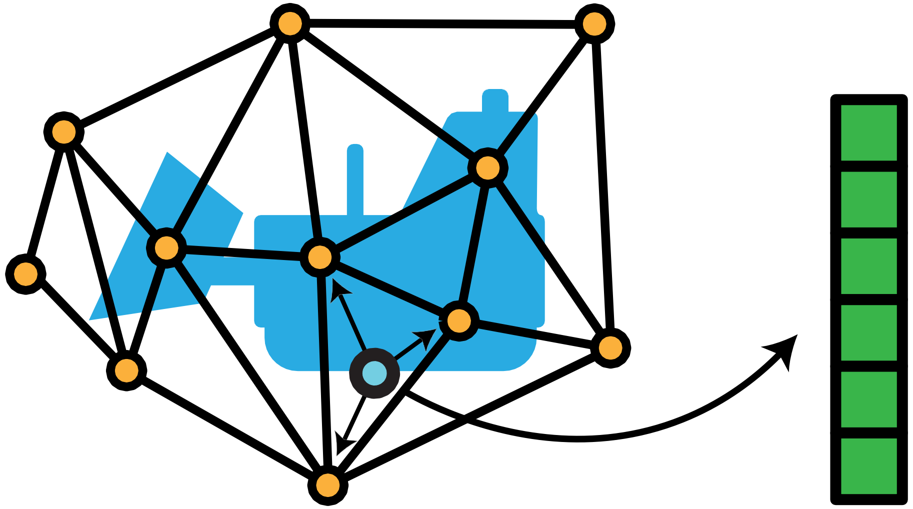
- Need a good initialization of point locations
- Redundancy
- Cannot produce high-quality surface

Mesh (Unstructured Grids)



[DefTet (Gao et al.), NeuralBody (Peng et al.), etc]

Mesh (Unstructured Grids)



[DefTet (Gao et al.), NeuralBody (Peng et al.), etc]

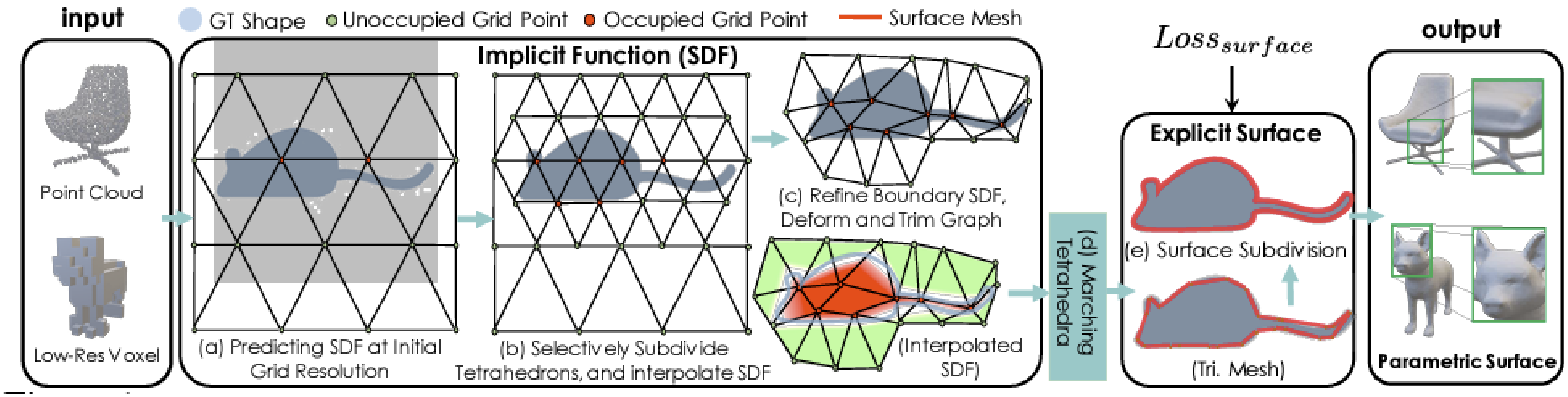
Pros:

- Can use the rich sets of tools in mesh processing
- Compatible with fast rendering

Cons:

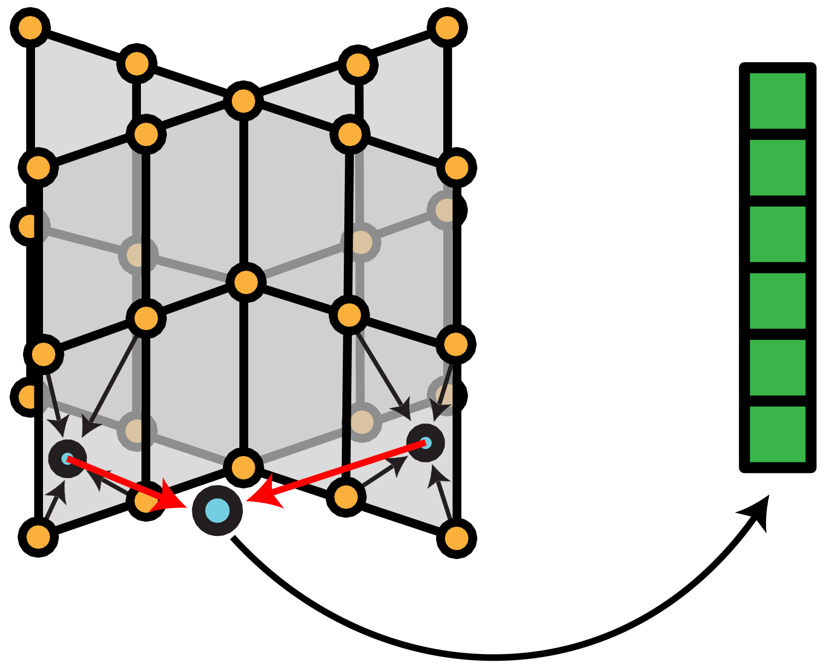
- Limited spatial resolution
- Hard to optimize

Deep Marching Tetrahedra (Shen et al.)



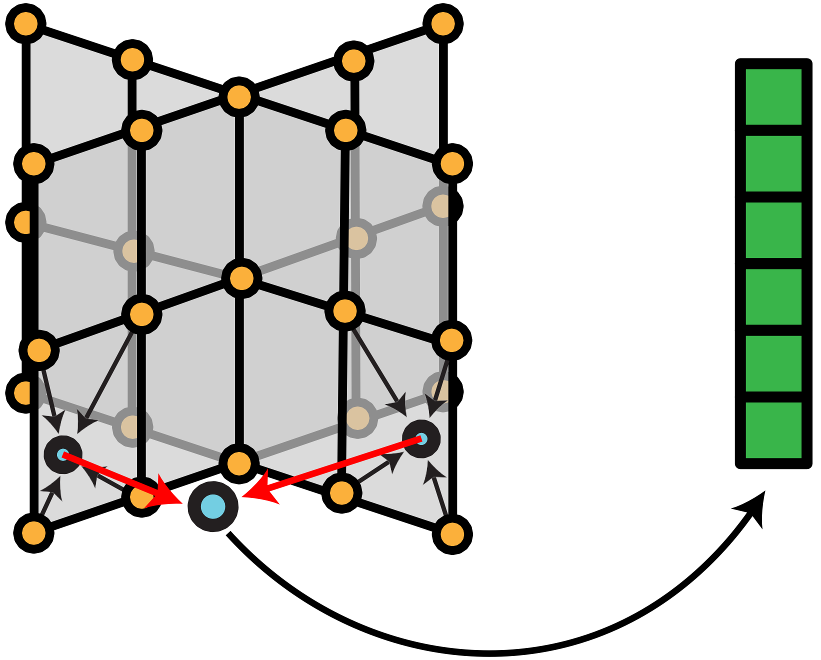
[Shen et al.]

Multiplanar Images



[Convolutional OccNet (Peng et al), EG3D (Chan et al.), etc]

Multiplanar Images



[Convolutional OccNet (Peng et al), EG3D (Chan et al.), etc]

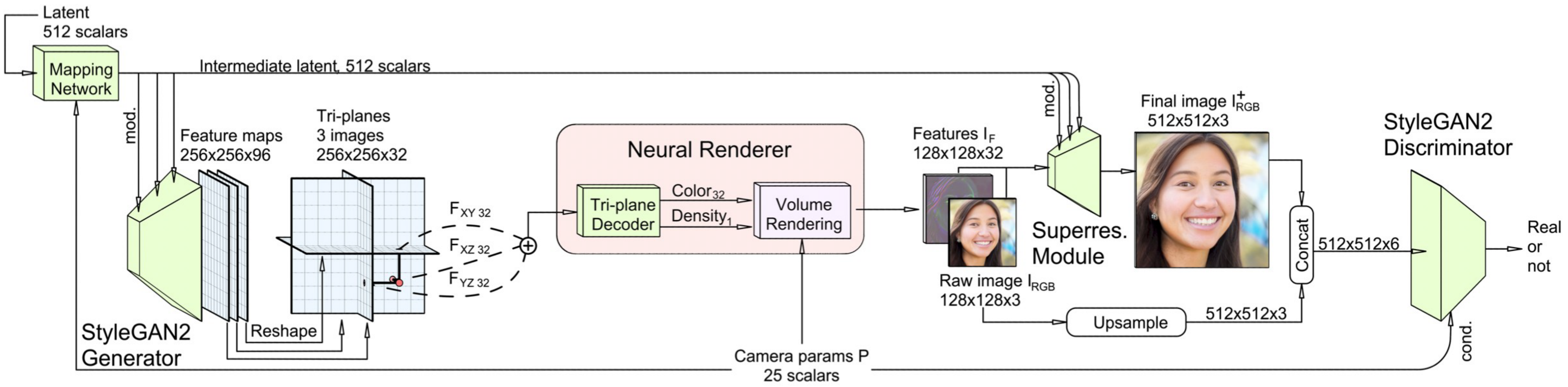
Pros:

- More compact than 3D dense grids
- Compatibility with 2D pipelines

Cons:

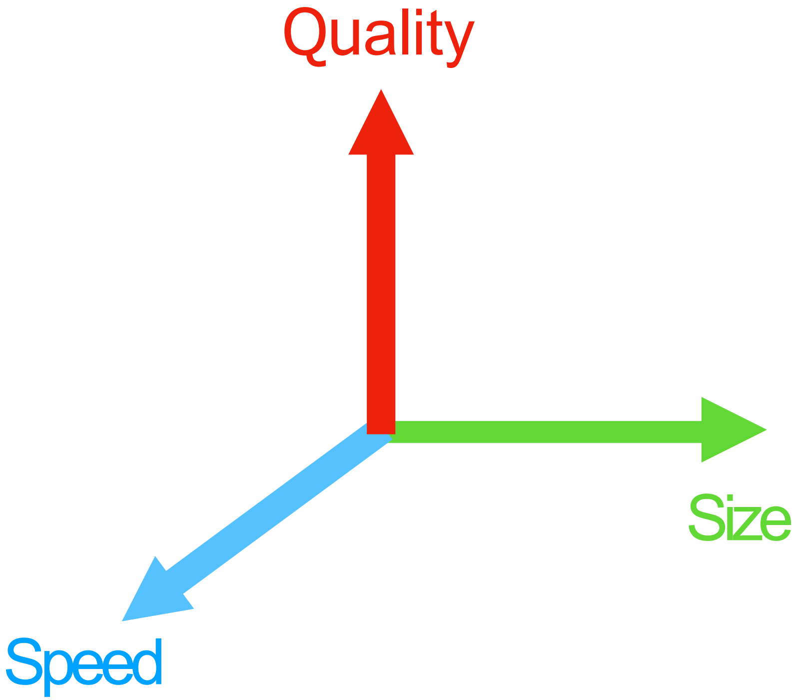
- Resolution bias on plane axis

EG3D (Chan et al.)

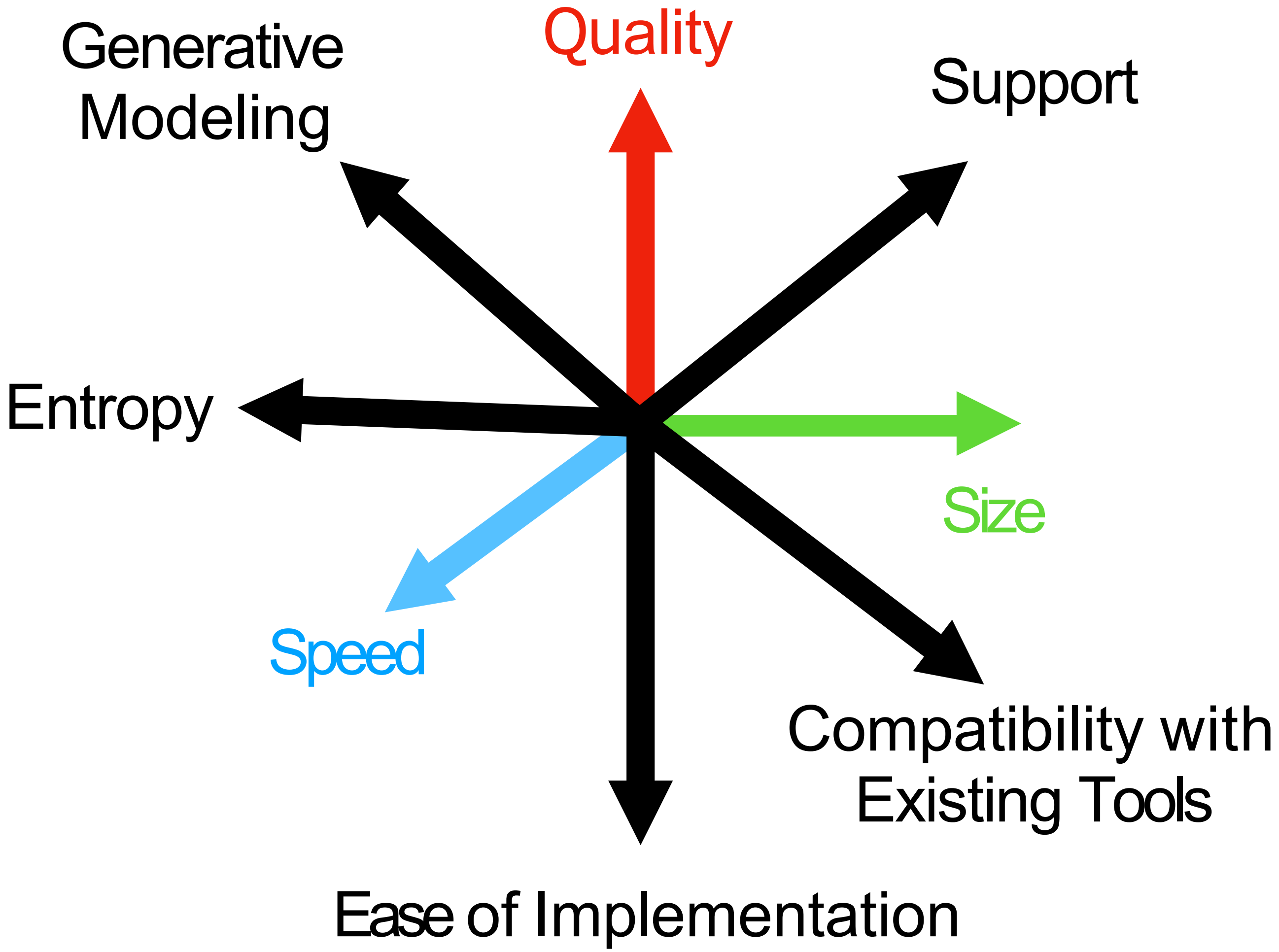


[Chan et al.]

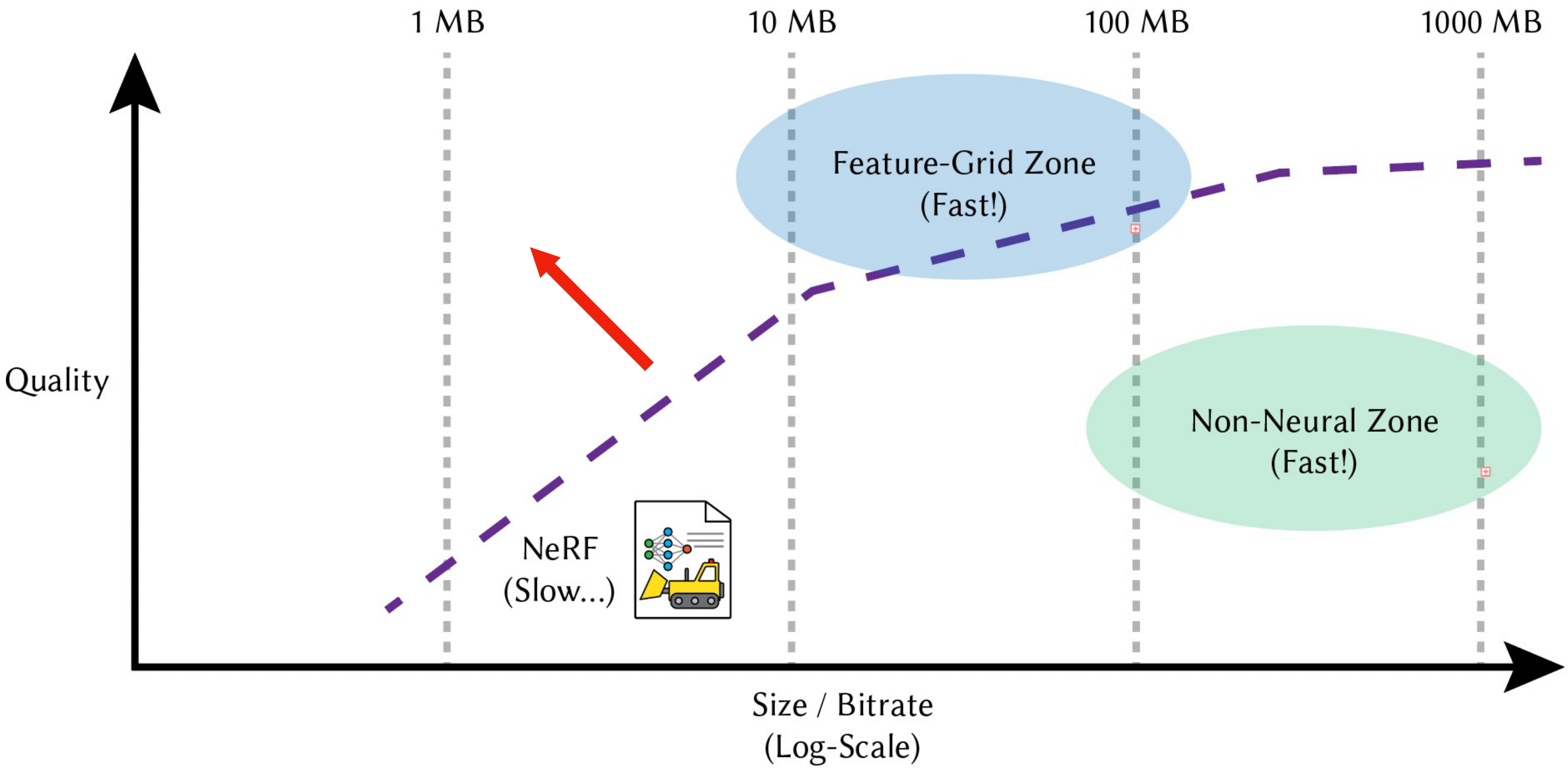
Design Tradeoffs



Design Tradeoffs



Design Tradeoffs



Acknowledgments

- Advances in Neural Rendering
- Neural Fields in Visual Computing and Beyond
- awesome-NeRF: a curated list of awesome neural radiance fields papers
- MPII Summer Semester 2023: Computer Vision and Machine Learning for Computer Graphics

Any Questions?