NeuralEditor: Editing Neural Radiance Fields via Manipulating Point Clouds

Jun-Kun Chen[†]   Jipeng Lyu[†]   Yu-Xiong Wang

Motivation and Contribution

- **Task**: edit shape of 3D scenes
  - Optimus Prime: humanoid ⇔ truck
  - Impact applications in visual industry
- **Objective**: render edited scenes
  - ✓ Visually faithful
  - ✓ Consistent with ambient environment

Shape Editing Task	Previous Work	Our NeuralEditor
Methodology	Deform viewing rays	Render on edited NeRF
Editing tasks	👎 Shape deformation only	👍 All shape editing tasks
Operation dexterity	👎 Simple, continuous, coarse	👍 Fine-grained, aggressive
Support fine-tuning	✗	✓

Key Insight

Implicit scene representation
- NeRF and its variants
- 👍 Good rendering results
- 👎 Cannot support shape editing

Explicit scene representation
- Point clouds, meshes, ...
- 👎 Do not render well
- 👍 Natively allow shape editing
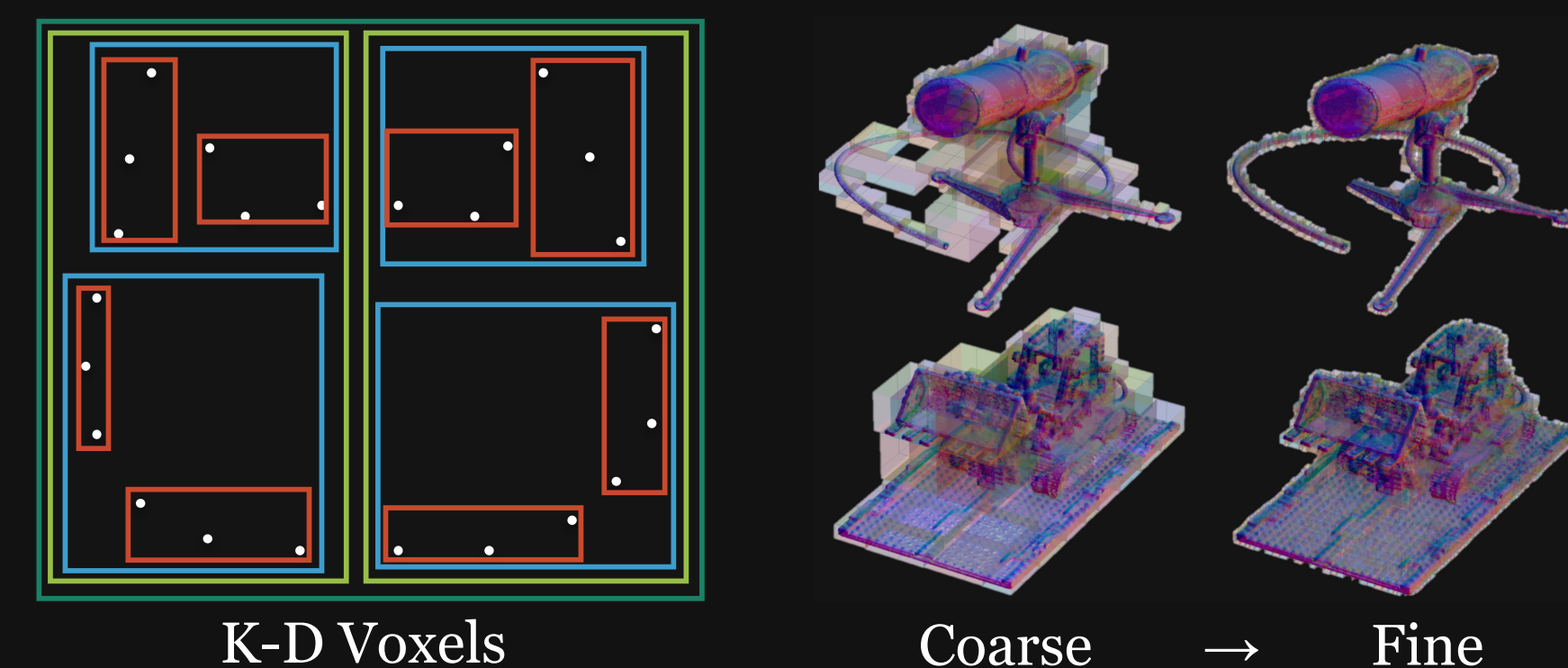
- ❖ Connection: NeRF rendering ≡ plotting a point cloud
- ➤ Our solution: point cloud-guided NeRF
  - Integrate both representations
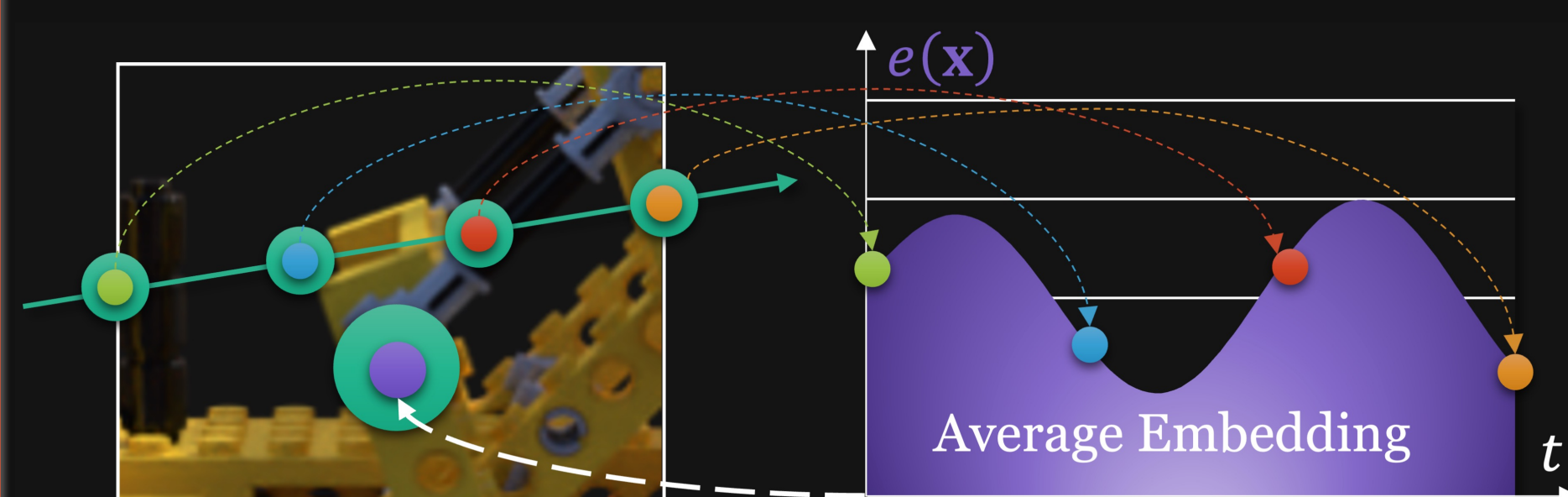 - Perform scene editing by manipulating its underlying point cloud

Point Cloud-Guided NeRF

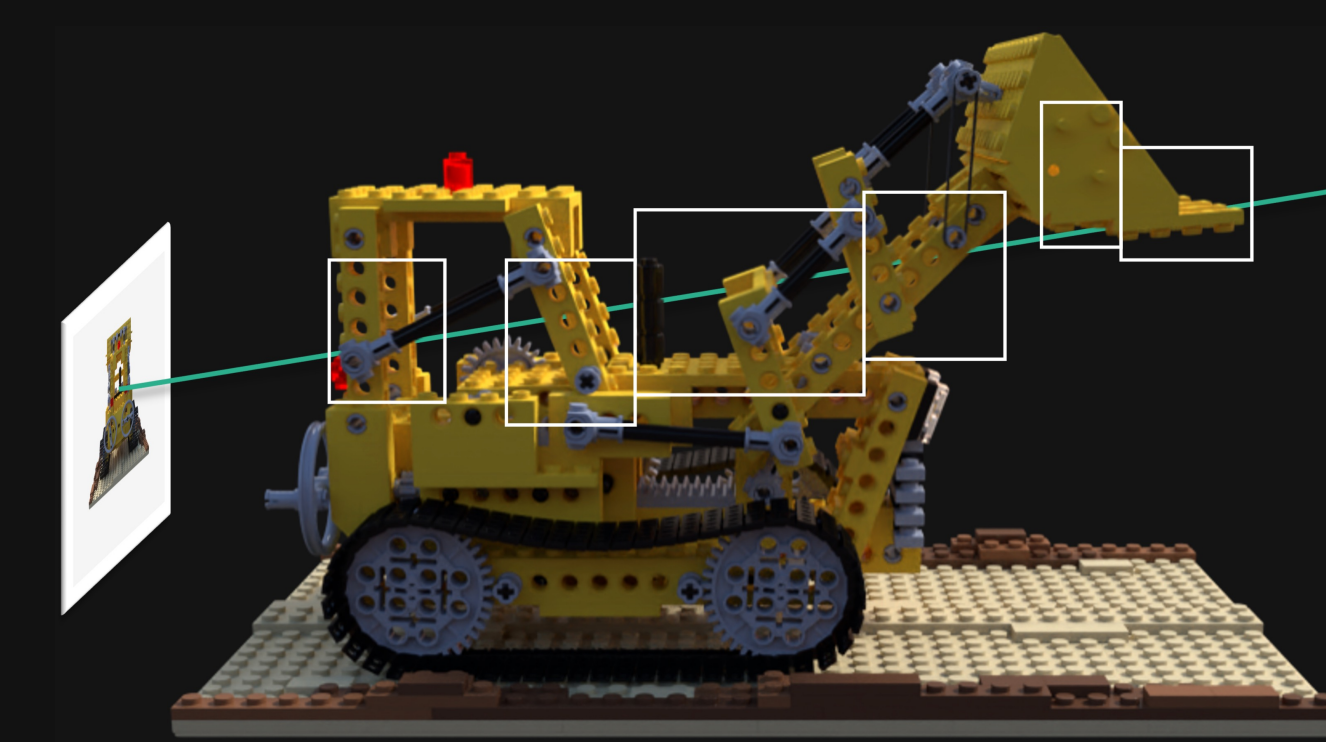## Novel Designs in Point Cloud-Guided NeRF

❖ **K-D Tree-Guided Voxels (K-D Voxels)**
- Multi-scale and shape-adaptive
- A native coarse-to-fine rendering guidance

K-D Voxels   Coarse → Fine

❖ **Render with K-D Voxels**
- Locate voxels via a top-down recursion
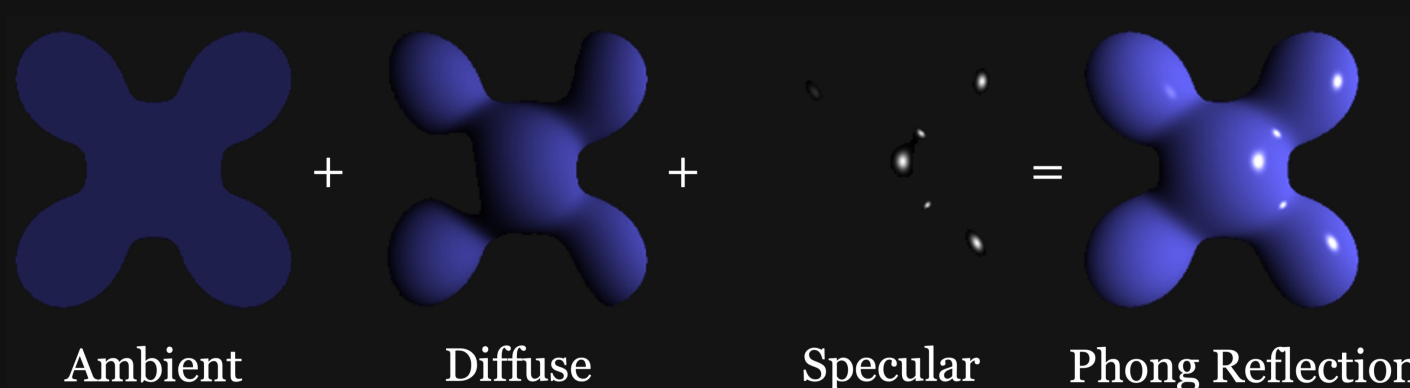- Focus on the boundary automatically

❖ **Deterministic Spline Integration**
- Apply deterministic spline integration, instead of random sampling
- Use average embedding as aggregation
- An efficient and stable approach

$e(\mathbf{x})$

Average Embedding t

❖ **Better Shape Modeling via Surface Norms**
- From point cloud: estimate from KNNs
- From NeRF: gradient of volume density
- Regularize to drive them close $\mathbf{n}, -\nabla\sigma$

❖ **Phong Reflection-Based Color Modeling**
- Utilize modeled norms in color modeling
- Better decomposition of color

Ambient + Diffuse + Specular = Phong Reflection

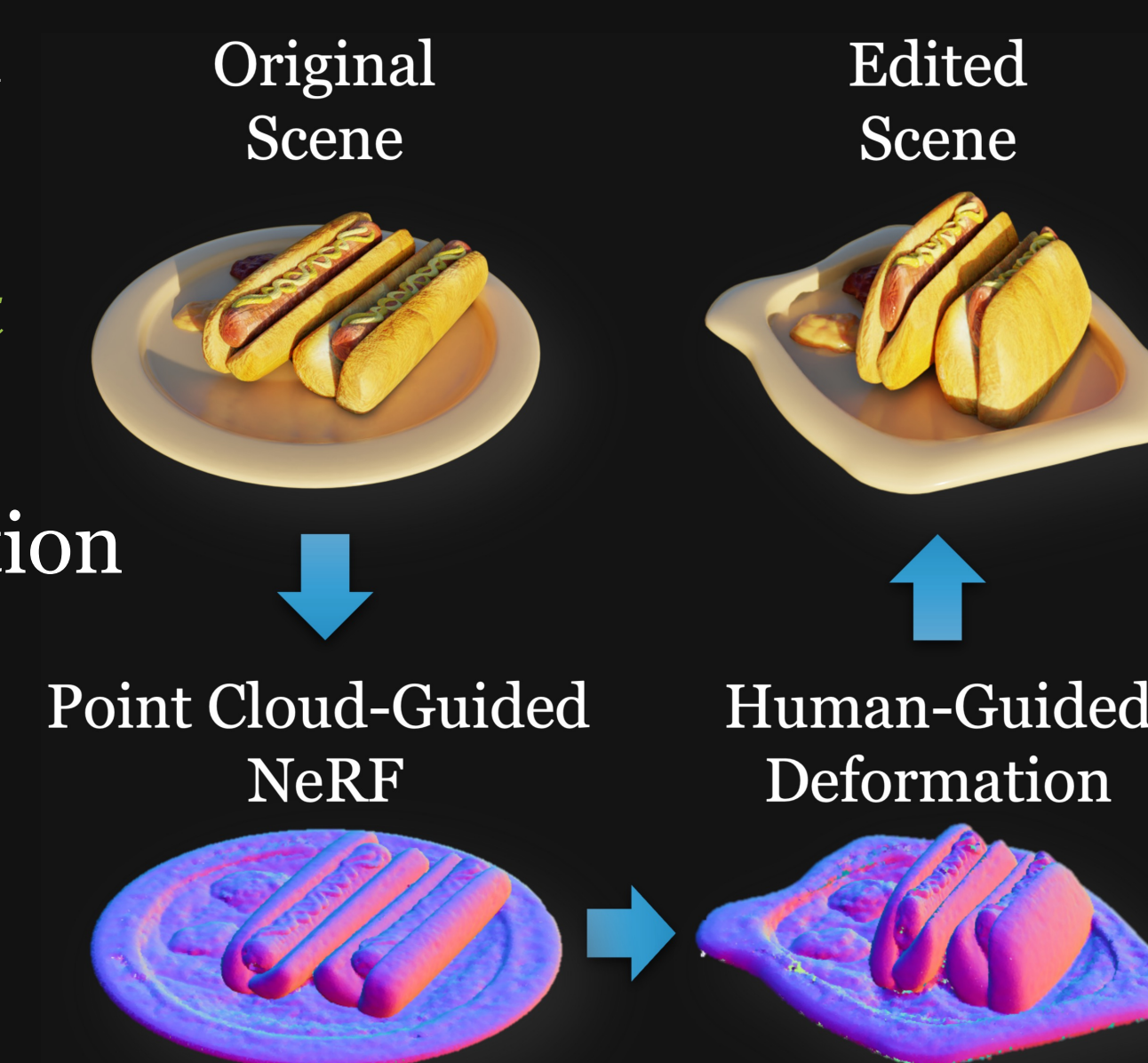## Unified Shape Editing Scheme

- **Unified** shape editing, defined with indexed point cloud
  - Point cloud: mapping from index to point, $i \to p_i$
  - Shape editing: another point cloud with same indices, $i \to q_i$
  - Editing: point of index i moves from 3D location p_i to q_i
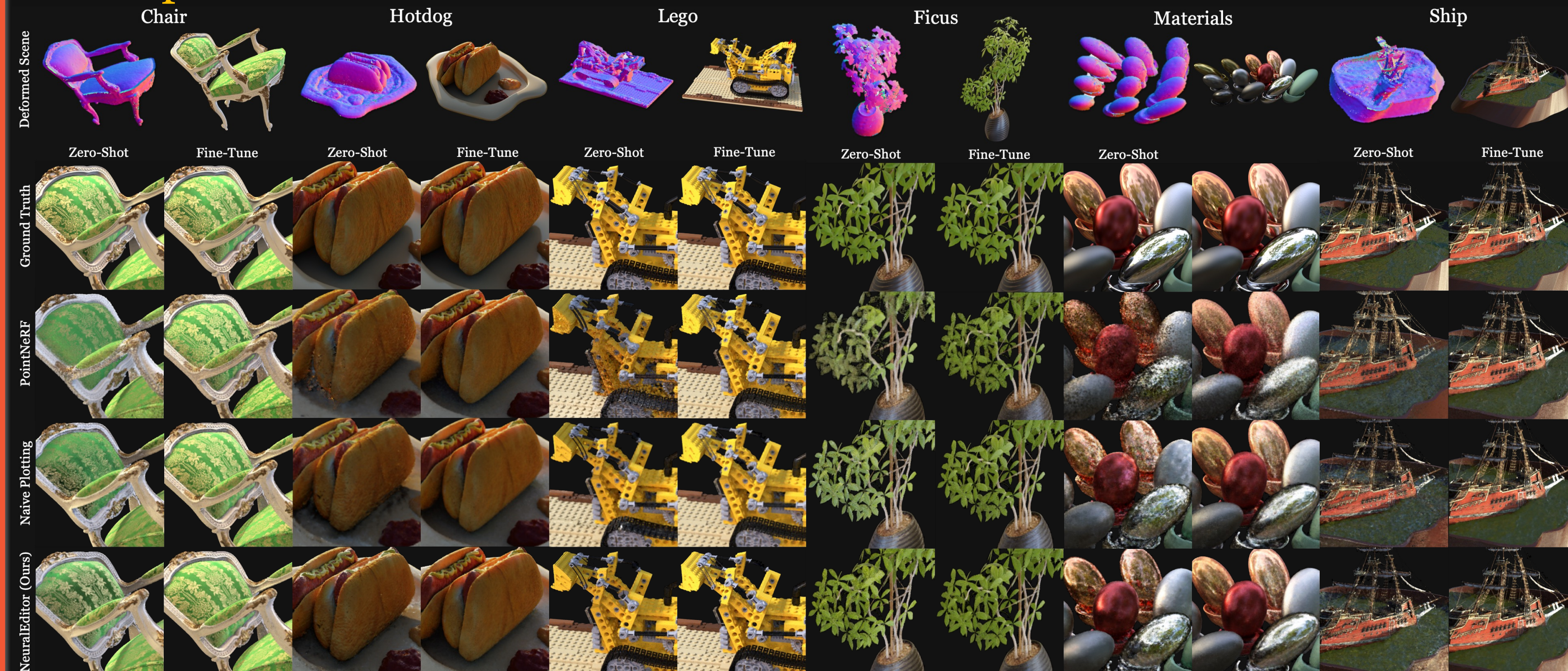- A simple yet general formulation with no extra assumption
  - 🚫 specific type of point motion 🚫 operation continuity
  - Support shape deformation, scene morphing, etc.
- Perform editing: replace the point cloud of our NeRF
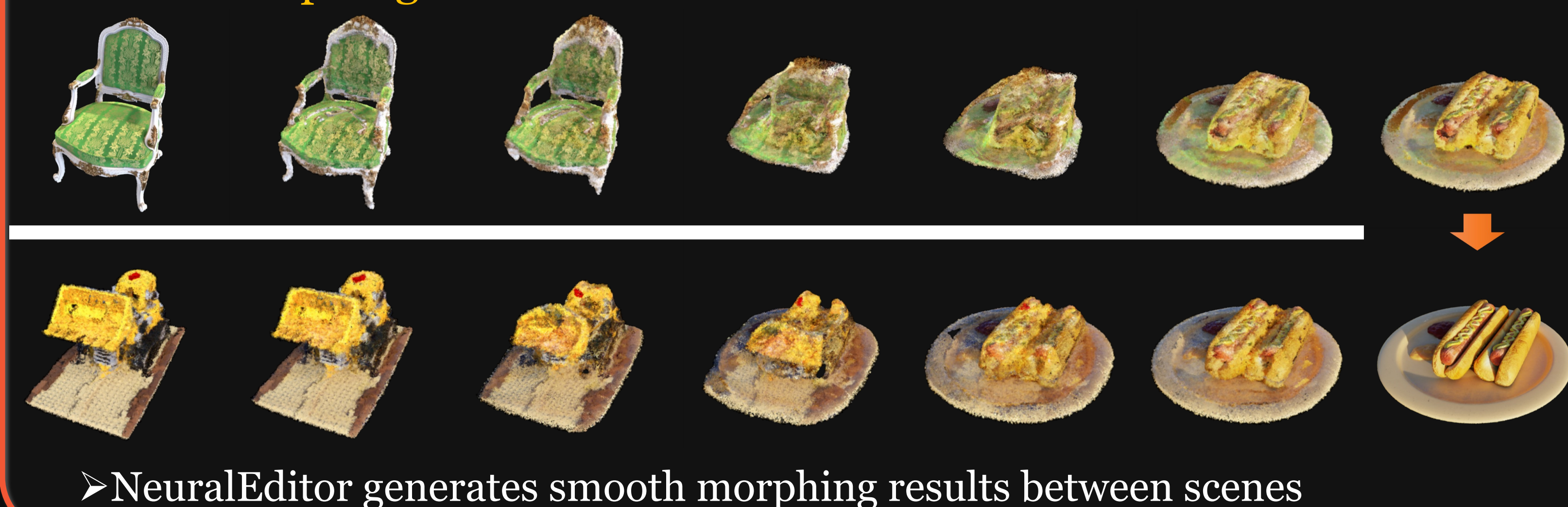 - ➤ Obtain a fully functional NeRF that supports fine-tuning

Original Scene   Edited Scene

Point Cloud-Guided NeRF   Human-Guided Deformation

Shape Editing Results

❖ **Shape Deformation Task**

Chair   Hotdog   Lego   Ficus   Materials   Ship

> Construct a novel shape deformation benchmark based on NeRF Synthetic
> NeuralEditor renders visually faithful results
> Further improve the consistency via fine-tuning
> Significantly outperform baselines quantitatively (up to 10% in PSNR)

❖ **Scene Morphing Task**

> NeuralEditor generates smooth morphing results between scenes

Conclusions

- **NeuralEditor** enables general shape editing on NeRF in a unified way
- **NeuralEditor** renders high-quality and visually faithful results in both shape deformation and scene morphing tasks

ArXiv   Presentation