NVIDIA

Data driven simulation for AV

Or Litany

ECCV
EUROPEAN CONFERENCE
ON COMPUTER VISION
TEL AVIV 2022

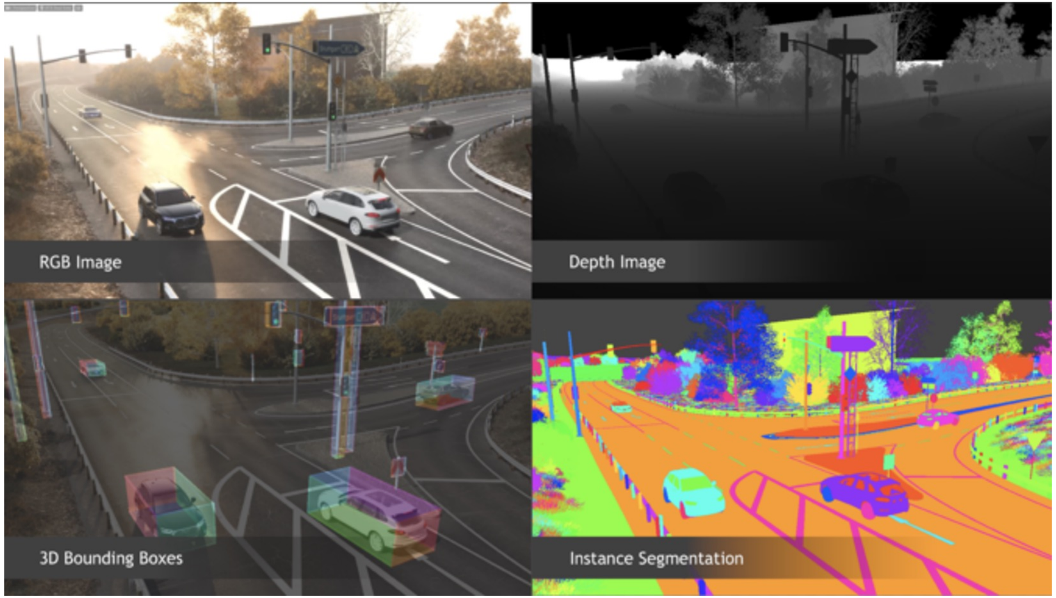
Traditional simulation

Pros:

- Generate training data
- Testing in a controlled environment

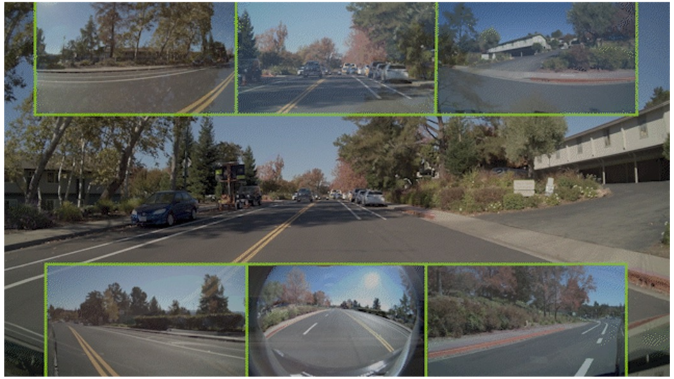
Limitations:

- Difficult to achieve realism
- Scale: Manual/rule-based content creation
- Log-replay testing: Open loop



Data-driven simulation

- **Content scale and diversity:** Harvest scenes, scenarios, and objects as simulation assets
- **Realism**: Mitigate the domain gap through sensor view synthesis
- **Closed-loop**: Learn traffic models from real world scenarios



→ Neural Sim →

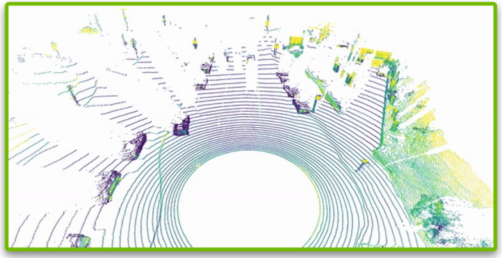
Building blocks

Scene reconstruction

Object reconstruction

Motion estimation and synthesis

Augmented reality



Sensor simulation

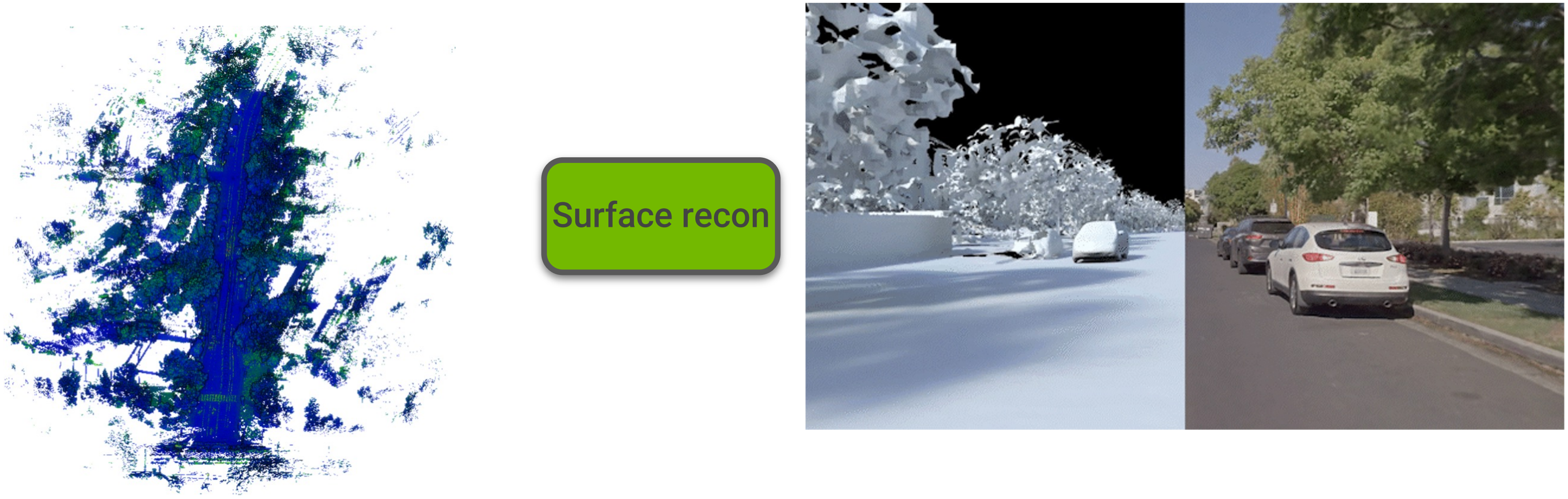
Scene reconstruction: Neural radiance field

- Remove dynamic objects, NeRF reconstruction

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

Scene reconstruction: Geometry

- Boosts novel view synthesis
- Provides drivable surface + environment for shadow casting



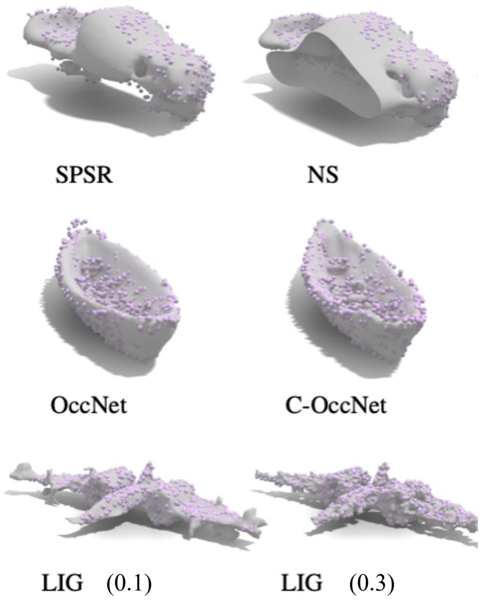
Neural kernel fields, Williams et al., CVPR 2022

Neural Kernel Fields for surface reconstruction in the wild

Data free: Respects input points, no extrapolations

Feed-forward: Miss details (“retrieval”)

Test-time optimization: slow, local minima



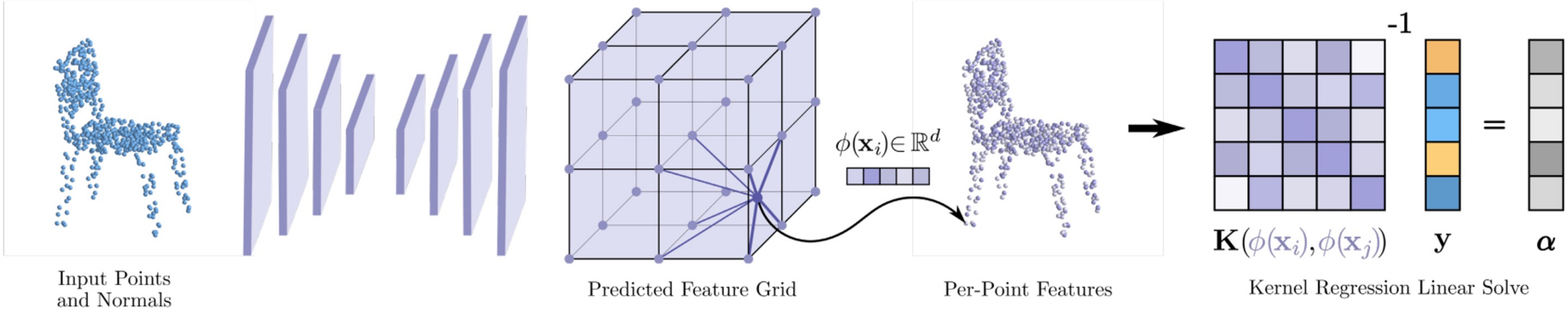
Neural kernel fields, Williams et al., CVPR 2022

Neural Kernel Fields for surface reconstruction in the wild

Learnable kernel:

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{x}'_j \in X'} \alpha_j K_{(\mathcal{X},\theta)}(\boldsymbol{x}, \boldsymbol{x}'_j).$$

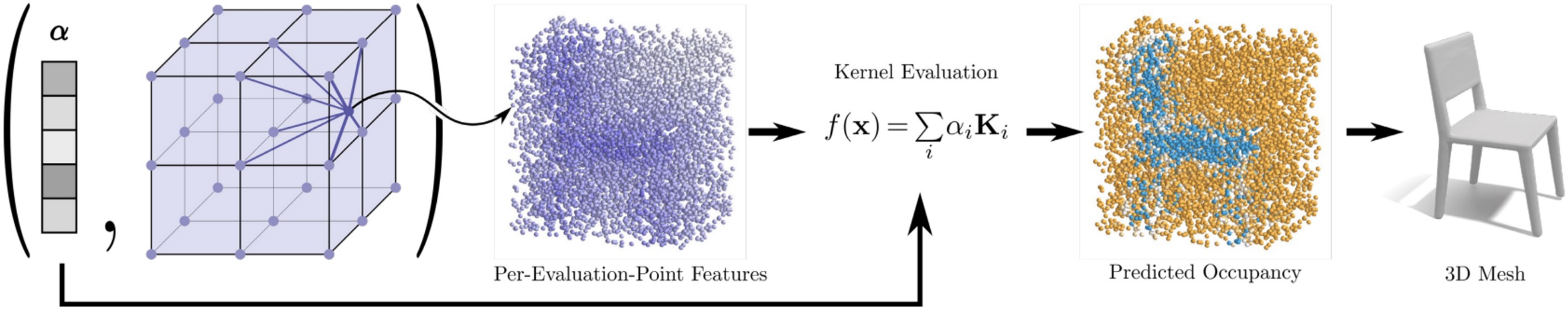
$$K_{(\mathcal{X},\theta)}(\boldsymbol{x}, \boldsymbol{z}) = K_{\text{NS}}([\boldsymbol{x} : \phi(\boldsymbol{x}|\mathcal{X}, \theta)], [\boldsymbol{z} : \phi(\boldsymbol{z}|\mathcal{X}, \theta)])$$



Neural kernel fields, Williams et al., CVPR 2022

Neural Kernel Fields for surface reconstruction in the wild

Evaluation:



Neural kernel fields, Williams et al., CVPR 2022

Neural Kernel Fields for surface reconstruction in the wild

Data free: Respects input points, no extrapolations

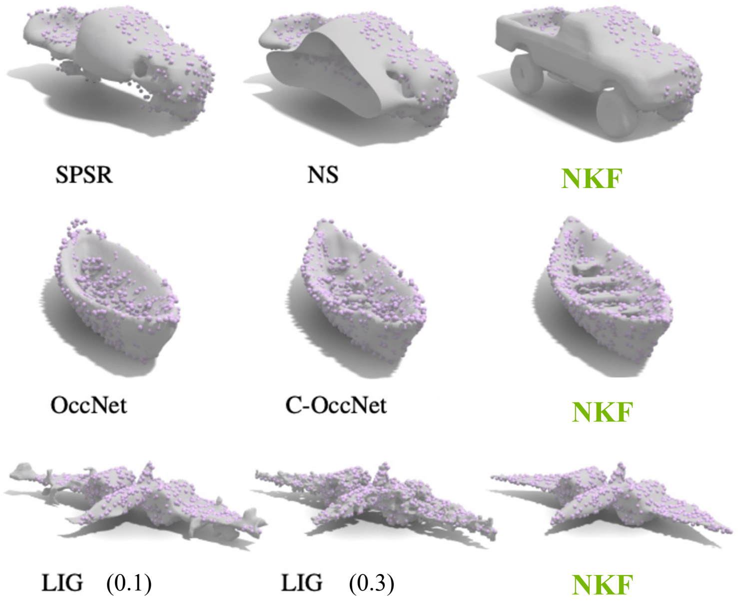
NKF: Data-driven priors

Feed-forward: Miss details (“retrieval”)

NKF: Linear test-time optimization to recover details

Test-time optimization: slow, local minima

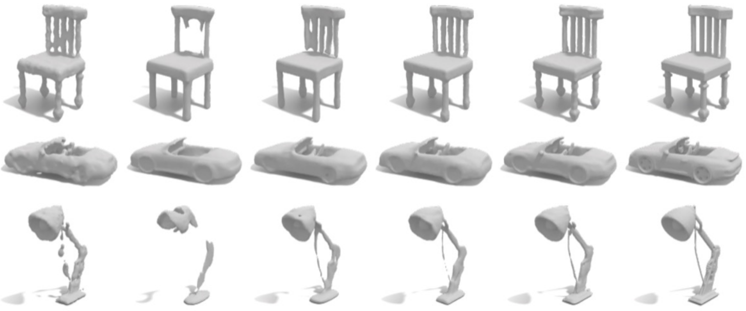
NKF: Global optimum



Neural kernel fields, Williams et al., CVPR 2022

In-category Reconstruction

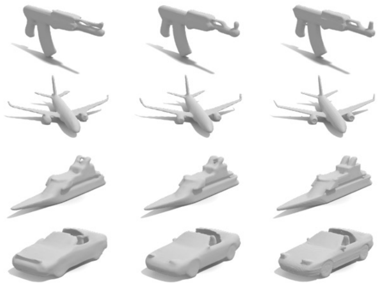
Noise	IoU ↑ $\sigma = 0.0$	$\sigma = 0.0025$	$\sigma = 0.005$
SPSR	0.772	0.759	0.735
OccNet	0.761	0.747	0.726
C-OccNet	0.828	0.848	0.857
NS	0.864	0.831	0.835
SAP	0.872	0.866	0.849
Ours	**0.947**	**0.908**	**0.866**
Ours w/o norm.	0.924	0.894	0.862



NS[4] OccNet[5] C-OccNet[6] SAP[8] Ours GT

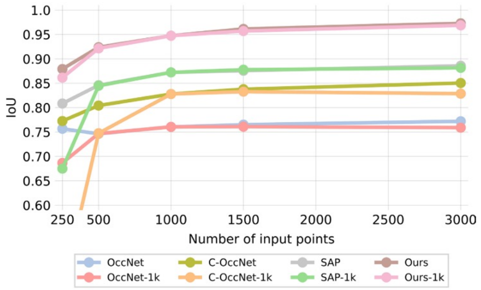
Generalization

Chair to Other Categories

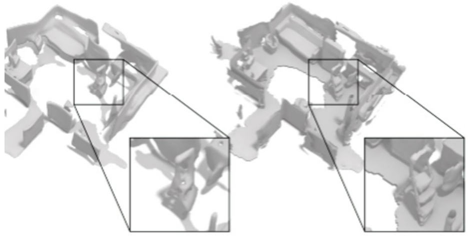


Ours (Chair) Ours GT

Across different number of points



ShapeNet[1] to ScanNet[2]



C-OccNet [48] **Ours**

Neural kernel fields, Williams et al., CVPR 2022

Next steps: Real-time + large scale

Work in progress

Asset harvesting

Goal: Reconstruct 3D assets from real-world driving data to be used in simulation.

Harvest assets from driving sequence

Insert harvested objects into a different scene

Extracting Triangular 3D Models, Materials, and Lighting From Images, Munkberg et al., CVPR 2022

Asset generation

GET3D: Trained from images

LION: Trained from pointclouds

GET3D: A Generative Model of High Quality 3D Textured Shapes Learned from Images, Gao et al., NeurIPS 2022
LION: Latent Point Diffusion Models for 3D Shape Generation, Zeng et al., NeurIPS 2022

Virtual object insertion

- Environment lighting estimation
- Create rare / safety-critical / hard-to-label scenarios.
- Comes with free labels (3D bbox, instance seg., etc.).

Neural Light Field Estimation for Street Scenes with Differentiable Virtual Object Insertion, Wang et al., ECCV 2022

Motion Estimation and Synthesis

- **Estimation & Replay:** Recover original motion from data and replay in simulation

Motion Estimation & Replay

STRIVE: Generating Useful Accident-Prone Driving Scenarios via a Learned Traffic Prior, Rempe et al., CVPR 2022

Motion Estimation and Synthesis

- **Synthesize & Edit:** Make simulation reactive, editable, and add new dynamic agents

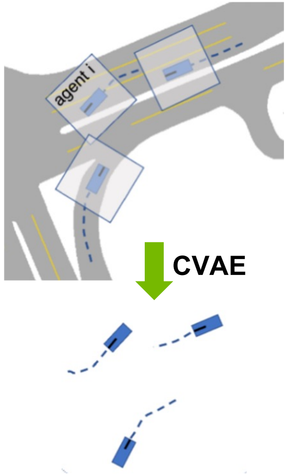
Motion Synthesis & Editing

STRIVE: Generating Useful Accident-Prone Driving Scenarios via a Learned Traffic Prior, Rempe et al., CVPR 2022

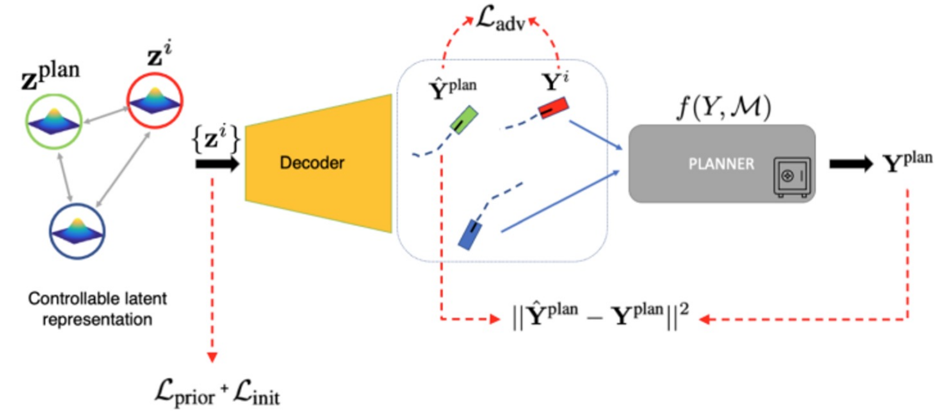
Challenging scenarios from real-world drives

Goal: recover vehicle motion from real-world data and modify to simulate challenging scenarios

1. Train a generative traffic model



- **In**: real-world driving dataset
- **Out**: future forecasting model for vehicle trajectories
- **How**: conditional VAE (CVAE)



2. Match planner

$$\min_{\mathbf{z}^{\text{plan}}} ||\hat{\mathbf{Y}}^{\text{plan}} - \mathbf{Y}^{\text{plan}}||^2 - \alpha \log p_\theta(\mathbf{z}^{\text{plan}} | X, \mathcal{M})$$

3. Adversarial Scenarios

$$\mathcal{L}_{\text{adv}} = \sum_{i=1}^{N} \sum_{t=1}^{T} \delta_t^i \cdot ||\mathbf{y}_t^i - \hat{\mathbf{y}}_t^{\text{plan}}||^2$$

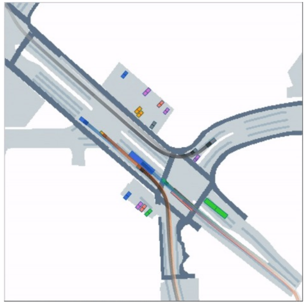
$$\delta_t^i = \frac{\exp(-||\mathbf{y}_t^i - \hat{\mathbf{y}}_t^{\text{plan}}||)}{\sum_j \sum_t \exp(-||\mathbf{y}_t^j - \hat{\mathbf{y}}_t^{\text{plan}}||)}$$

STRIVE: Generating Useful Accident-Prone Driving Scenarios via a Learned Traffic Prior, Rempe et al., CVPR 2022

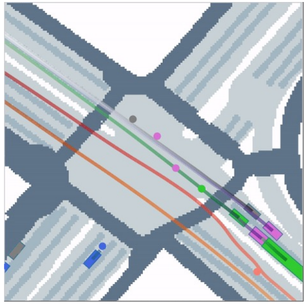
Next: Controllability

Goal: Meet user-defined constraints at simulation time

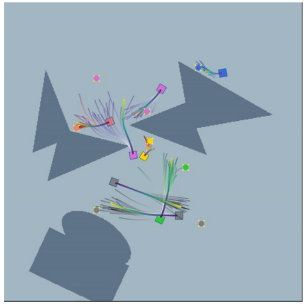
Avoid Collisions



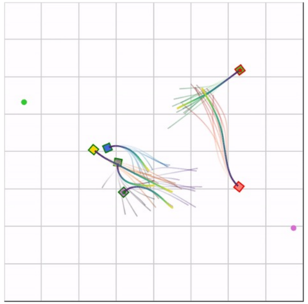
Follow Waypoints

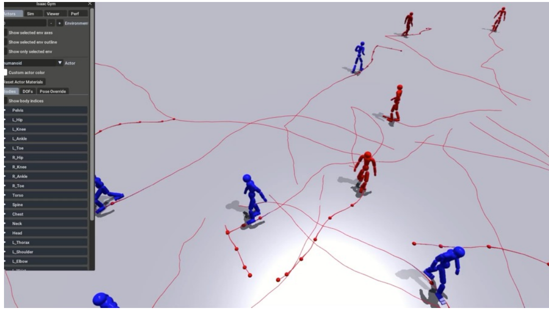


Obstacle Avoidance



Social Groups





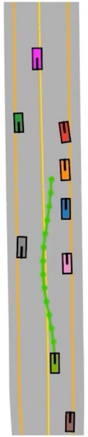
Vehicles

Pedestrians

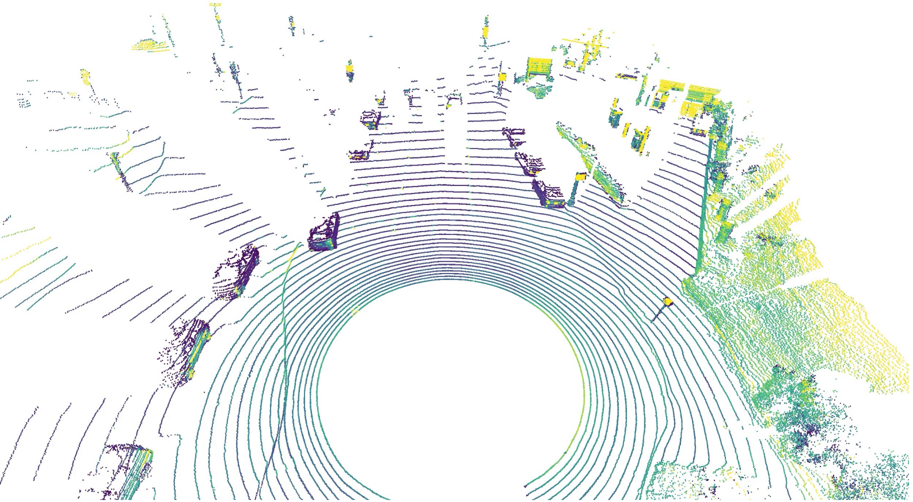
Work in progress

Active sensors resimulation

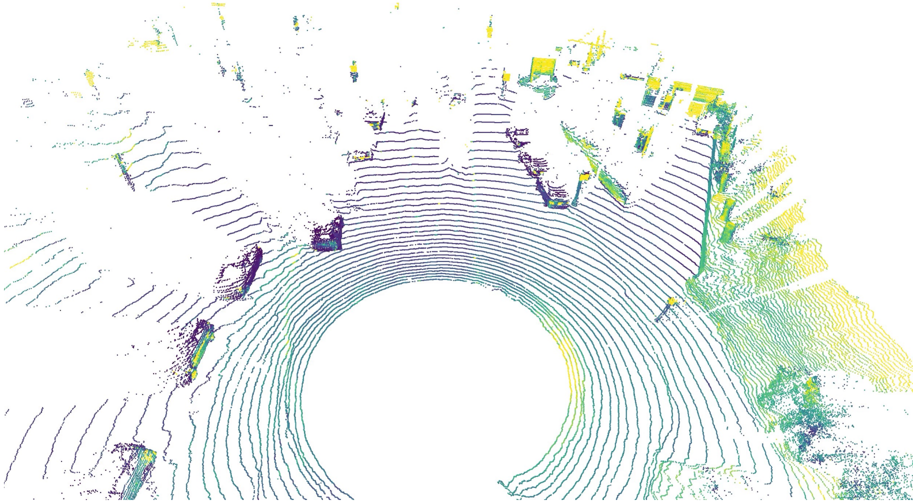
- Perception systems often rely on **active** sensors (e.g. LIDAR) in addition to camera
- Reenactment requires novel view synthesis of these sensors
- Baseline explicit approach:
  - Step 1: Reconstruct a 3D surface from sparse pointclouds
  - Step 2: Learn ray-dropping



Active sensors with neural fields



Original sensor data



LiDAR sensor shifted left by 1 m

Summary

- Useful simulation: Scale, visual and behavioural realism, closed-loop
- These can be achieved in a data-driven fashion:
  - Scene and object reconstruction (color, geometry, material properties)
  - Controllable traffic + motion models
  - Sensor simulation
- Future direction: Controllability, generalization