Robot Perception & Intelligence

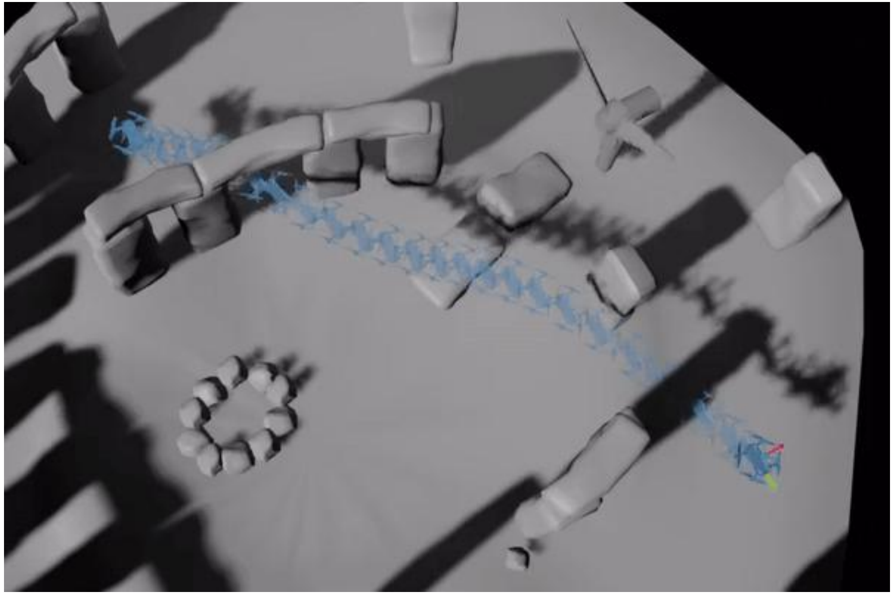
Robot Navigation in Neural Radiance Fields

Student: Yudong Nan

Advisor: Simon Boche

Garching, 02. December 2024

Motivation

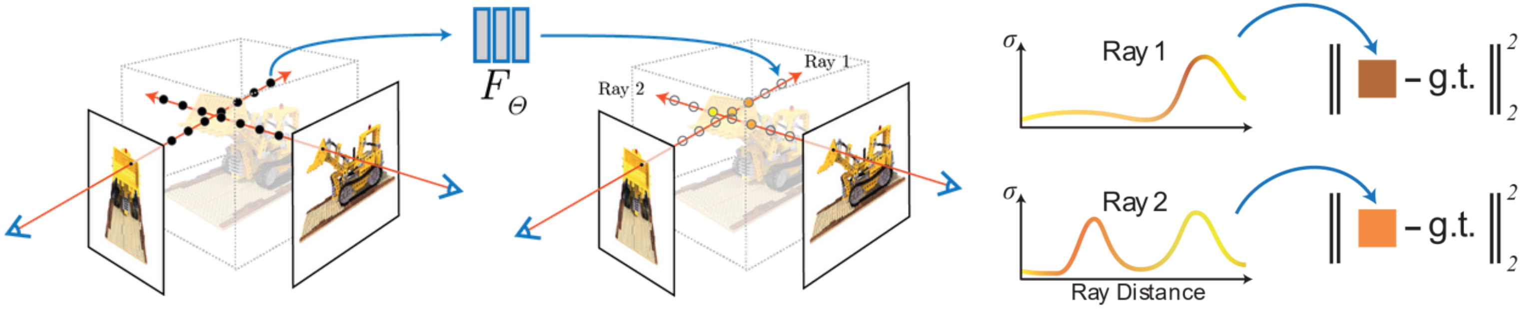


Source: Mip-NeRF 360 [Barron 2022], NeRF-Nav [Adamkiewicz 2022]

Outline

Neural Radiance Fields (NeRF)

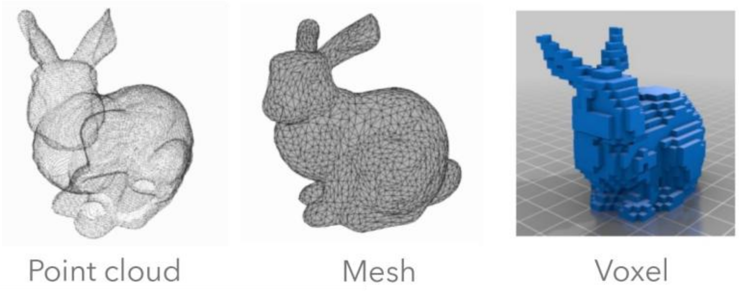
$$(x,y,z,\theta,\phi) \rightarrow F_{\Theta} \rightarrow (RGB\sigma)$$

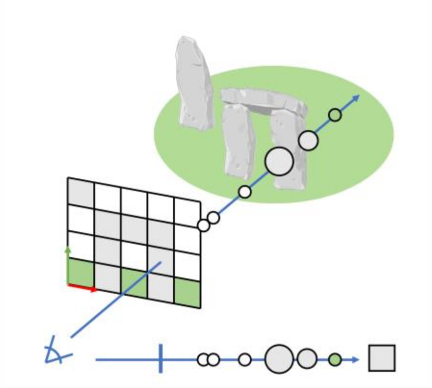


NeRF as environment representation

Advantages:

- Direct training on RGB images
- Continuous density field, inherently encodes uncertainty
- Photorealistic synthetic images of unseen parts
- Captures complex materials (smoke, water, etc.)
- Memory efficiency for complex scenes





Challenge:

- Cannot directly provide spatial occupancy.
- Difficulty in estimating collision probabilities.

Related Works

IEEE ROBOTICS AND AUTOMATION LETTERS. PREPRINT VERSION. ACCEPTED JANUARY, 2022.

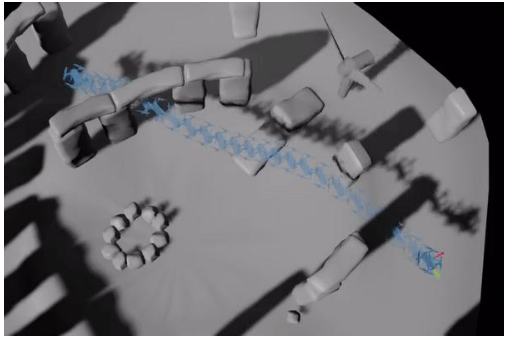
Vision-Only Robot Navigation in a Neural Radiance World

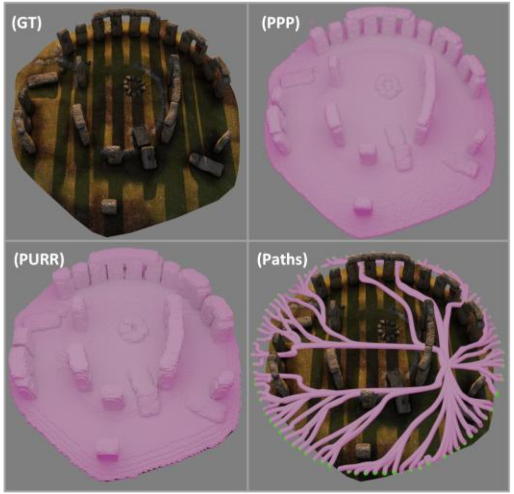
Michal Adamkiewicz,*[1] Timothy Chen,*[2] Adam Caccavale,[3] Rachel Gardner,[1] Preston Culbertson,[3] Jeannette Bohg,[1] Mac Schwager[2]

2712 IEEE TRANSACTIONS ON ROBOTICS, VOL. 40, 2024

CATNIPS: Collision Avoidance Through Neural Implicit Probabilistic Scenes

Timothy Chen, Preston Culbertson, *Member, IEEE*, and Mac Schwager, *Member, IEEE*





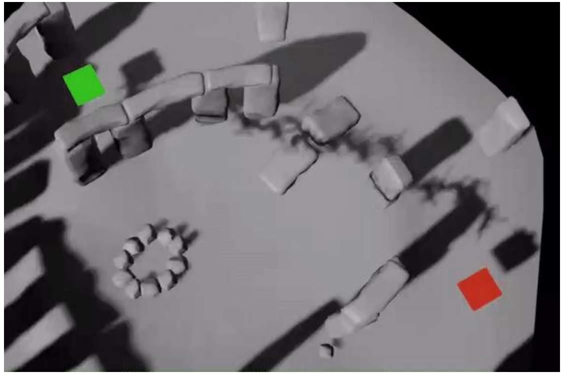
Source: NeRF-Nav [Adamkiewicz 2022], CATNIPS [Chen 2024]

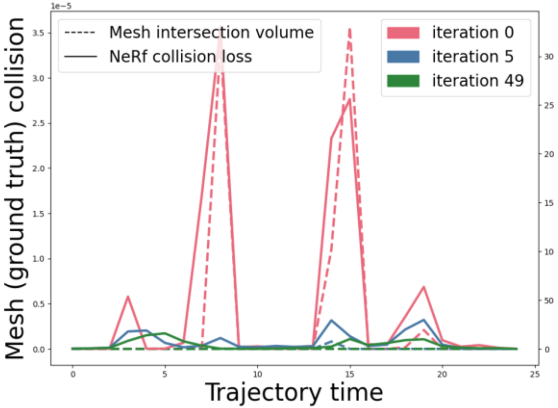
NeRF-Nav: Gradient-based Trajectory Optimization

$$p_t^{\text{coll}} = P\left(\bigcup_{\mathbf{b}_t \in \mathcal{B}} \mathbf{b}_t \in \mathcal{X}_{\text{coll}}\right) \geq \sum_{\mathbf{b}_t \in \mathcal{B}} \rho(\mathbf{b}_t)\, s(\mathbf{b}_t) \tag{1}$$

$$J(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_h) = \sum_{\tau=0}^{h} \left[\overbrace{\sum_{\mathbf{b}_i \in \mathcal{B}} \rho\left(\mathbf{R}_\tau \mathbf{b}_i + \mathbf{p}(\boldsymbol{\sigma}_\tau)\right) s(\mathbf{b}_i)}^{\text{collision penalty}} + \overbrace{\mathbf{u}_\tau^T \boldsymbol{\Gamma} \mathbf{u}_\tau}^{\text{control penalty}} \right] \tag{2}$$

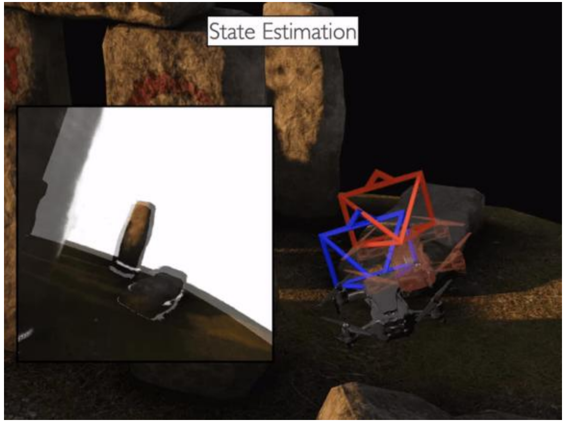
$\mathbf{b}_t \in \mathcal{B}$: point from robot bounding box,
$\mathcal{X}_{\text{coll}}$: collision set,
$\rho(\mathbf{b}_t)$: NeRF density value of point $\mathbf{b}_t$,
$s(\mathbf{b}_t)$: distance traveled by a body-fixed point $\mathbf{b}_t$,
$\mathbf{u}_\tau$: control input at time τ,
$\boldsymbol{\Gamma}$: diagonal matrix of weights penalizing control effort.





NeRF-Nav: Optimization-based State Estimation

$$J(\boldsymbol{\mu}_t) = \overbrace{\|C_{\mathcal{J}}(\mathbf{T}_t) - I_t(\mathcal{J})\|^2_{\mathbf{S}_t^{-1}}}^{\text{photometric loss}} + \overbrace{\|\boldsymbol{\mu}_{t|t-1} - \boldsymbol{\mu}_t\|^2_{\boldsymbol{\Sigma}_{t|t-1}^{-1}}}^{\text{process loss}} \quad (3)$$



$\boldsymbol{\mu}_t$: robot differential flatness state,

$C_{\mathcal{J}}(\mathbf{T}_t)$: rendered image from NeRF,

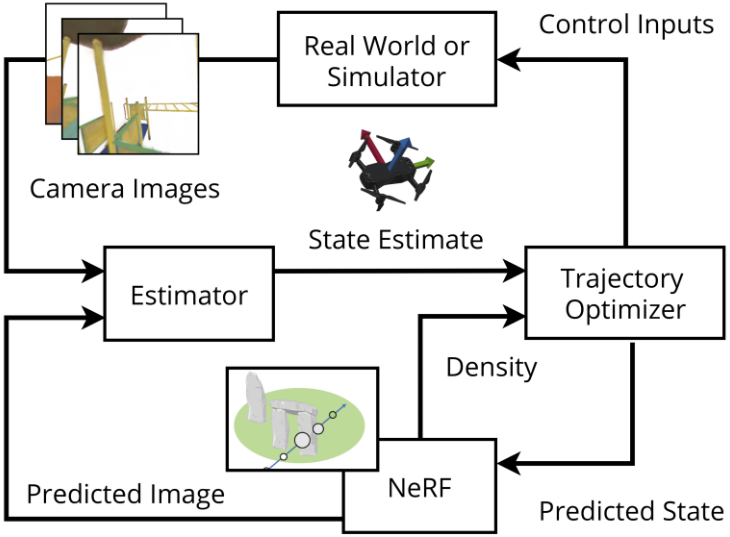
$I_t(\mathcal{J})$: real image pixels subject to ORB feature detector,

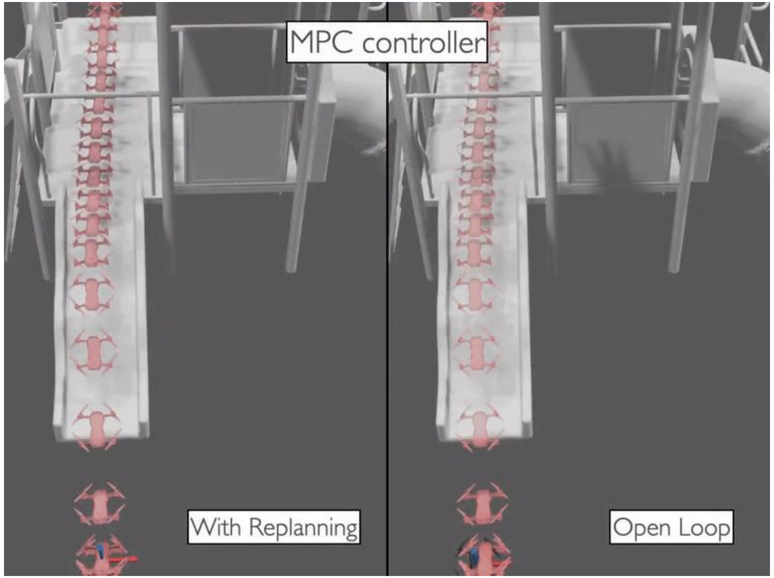
$\mathbf{S}_t$: measurement noise covariance,

$\boldsymbol{\Sigma}_{t|t-1}$: robot state covariance,

$\|x\|^2_M = x^T M x$ is the weighted l_2 norm.

NeRF-Nav: Online Replanning for Vision-only Navigation

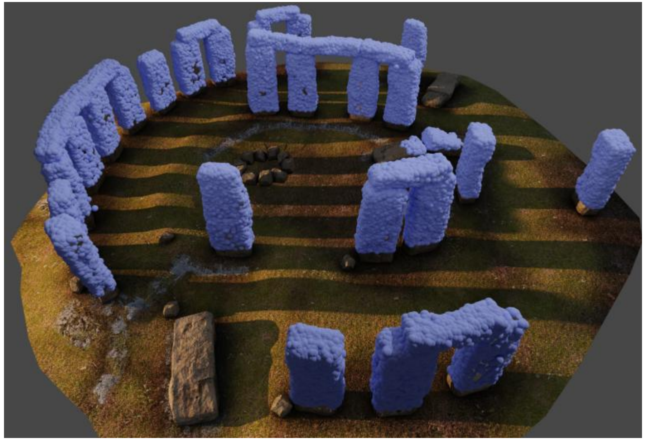




CATNIPS: NeRF Density as a Poisson Point Process (PPP)

Under PPP Smoothness Assumptions,

NeRF density ρ(p)	PPP intensity λ(x)
NeRF radiance/color	PPP marking
NeRF volumetric rendering	Compute expected pixel color of a marked PPP model
Training the NeRF	Ditting the PPP intensity param λ(x)

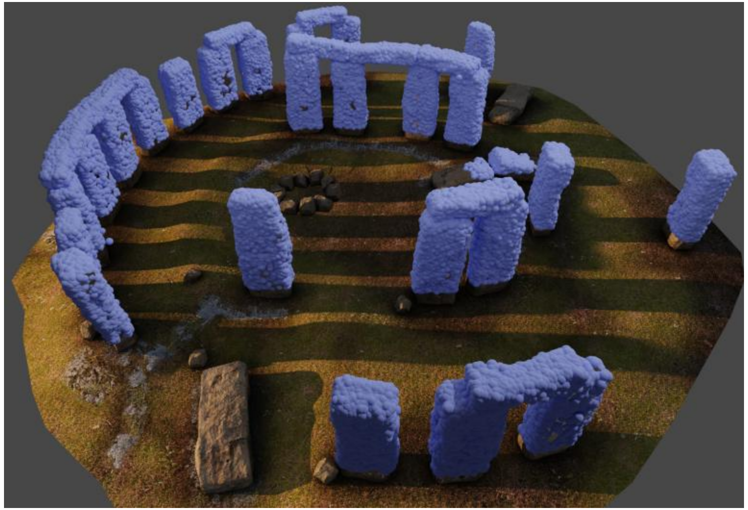


Collision Probability → $Pr(N(B(\mathbf{p}, \mathbf{R})) \leq N_{\text{aux}}^{\max}; \Lambda_B) \geq \sigma$

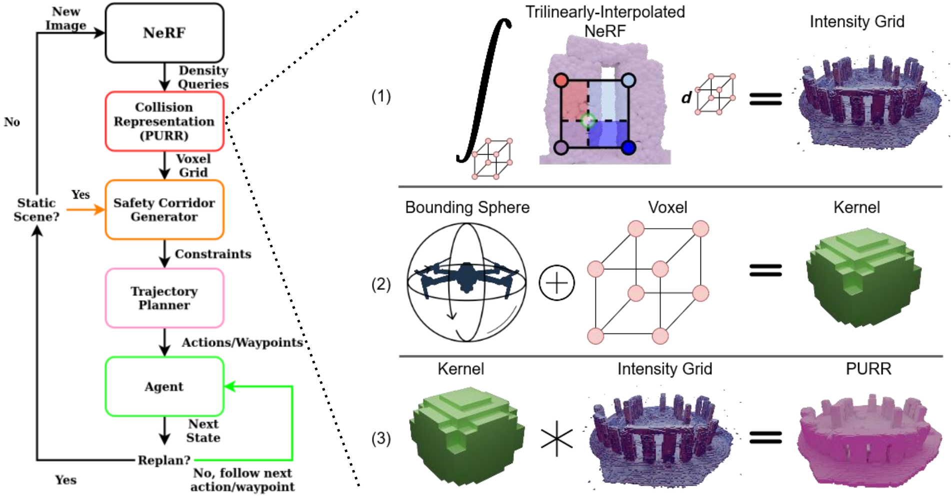
Source: CATNIPS [Chen 2024]

CATNIPS: NeRF Density as a Poisson Point Process (PPP)

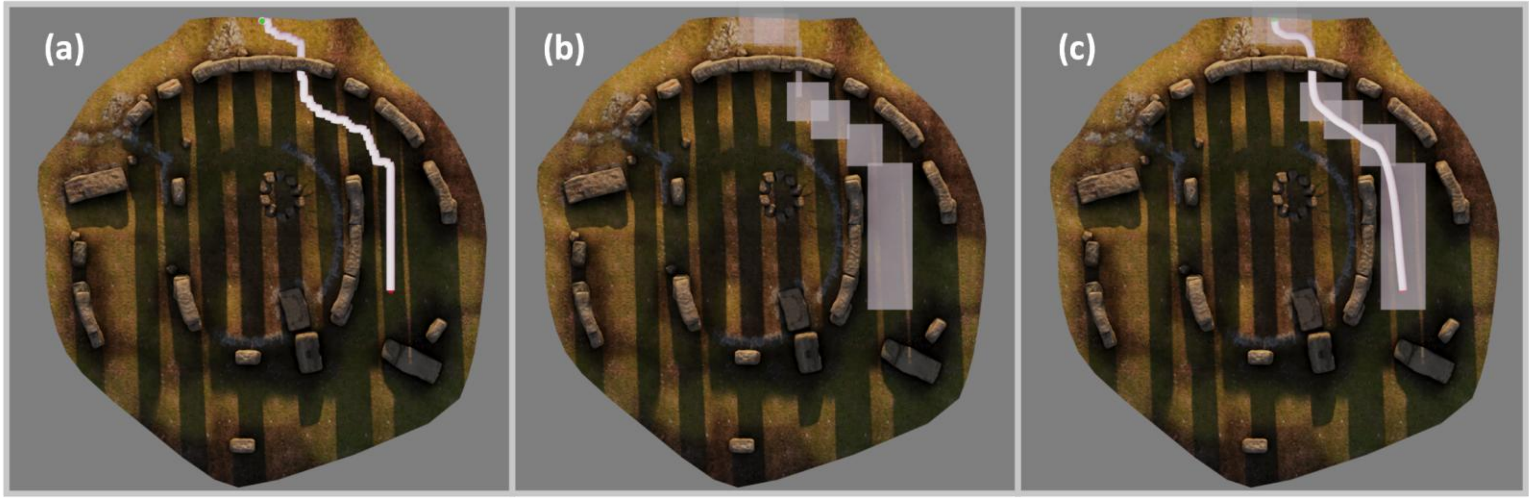
Probabilistic Formulation: PPP



Probabilistic Unsafe Robot Region: PURR



CATNIPS: Chance-Constrained Trajectory Generation

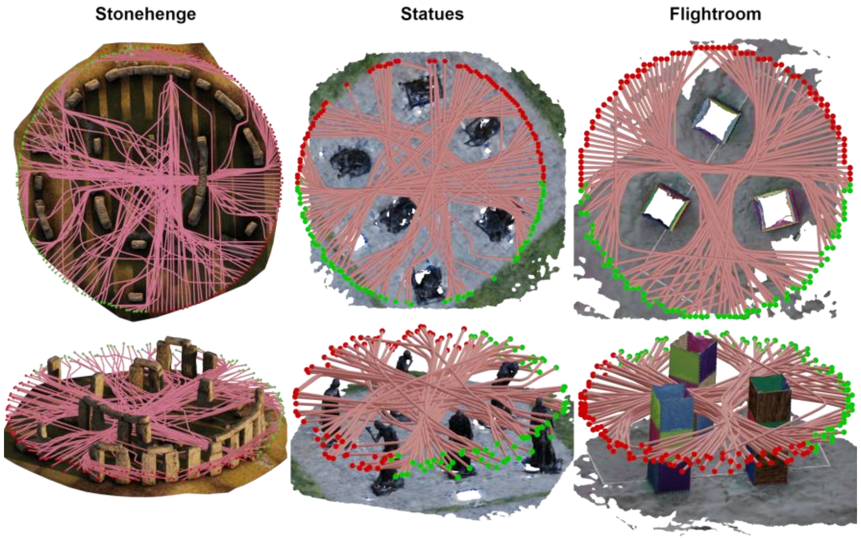


Initial A* Search → Bounding Box Generation → Smooth Trajectory Generation via Bézier Curves

Source: CATNIPS [Chen 2024]

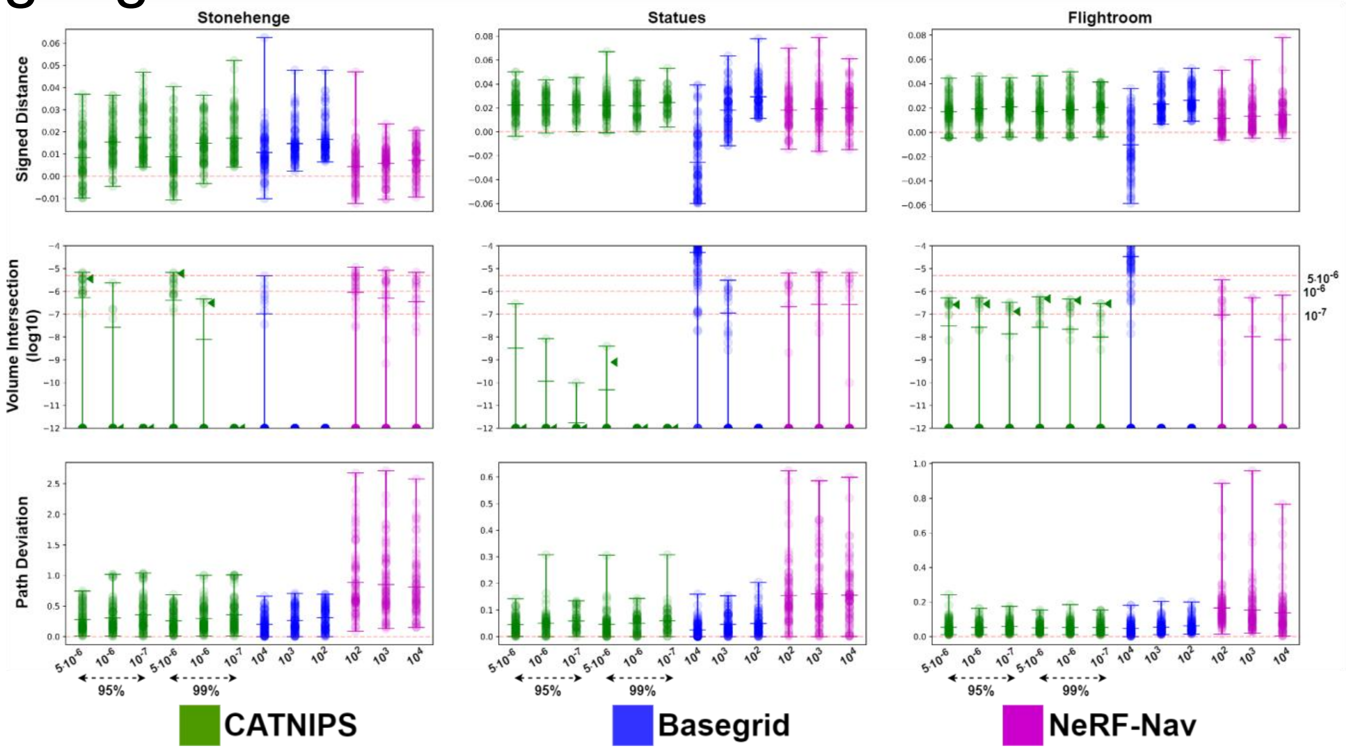
Experimental Setup

- CATNIPS vs. Basegrid vs. NeRF-Nav with same A* initialization
- Metrics:
  - Safety: Signed distance, Volume intersection
  - Conservativeness: Path deviation
  - Efficiency: Computation time
- Planning on 100 randomized configurations
- Hardware: RTX 4060 Laptop GPU



Source: CATNIPS [Chen 2024]

Planning Algorithm Performance



Computation Times

Computation Time (seconds)		
Operation	CATNIPS/Basegrid	NeRF-Nav
Offline		
Robot Kernel	0.002	0.002
PURR/Basegrid	1.11	1.11
Gradients	N/A	9.31*
Online		
A*	0.16 ± 0.05	0.16
Bounding Box	0.12 ± 0.09	N/A
B-Spline	0.034 ± 0.029	N/A
Gradients	N/A	0.93**

* 1000 gradient steps.
** 100 gradient steps.

Source: CATNIPS [Chen 2024]

Extensions to NeRF itself

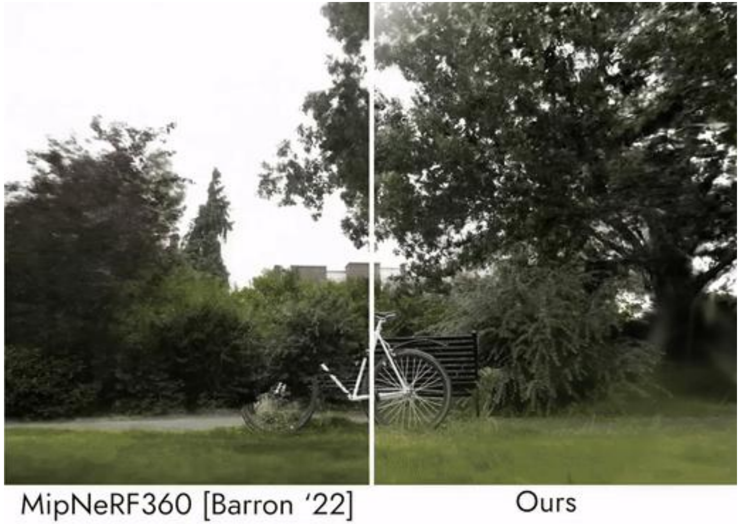
NeRF Limitations:

- Weak generalization ability across scenes
- Focus mainly on static environments and tasks
- Poor real-time performance: NeRF renders pixel-by-pixel, large rendering and training

New NeRF variants:

- Rapid scene adaptation using a prior over scenes [Tancik 2021]
- Handling dynamic scenes and moving objects, eg. D-NeRF [Pumarola 2021]
- Real-time efficiency, e.g. FastNeRF [Garbin 2021], InstantNGP [Müller 2022]
- 3D Gaussian Splatting?

Towards 3D Gaussian Splatting?

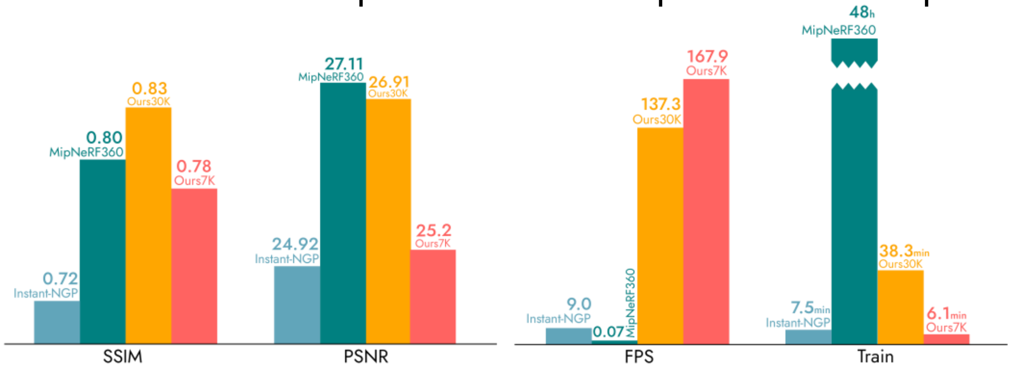


3D Gaussian Splatting:

- Smooth & differentiable
- Explicit geometry

Compared with NeRFs:

- State-of-the-art visual quality with real-time rendering (≥100 FPS at 1080p resolution)
- Limited interpolation at blank space or unseen parts

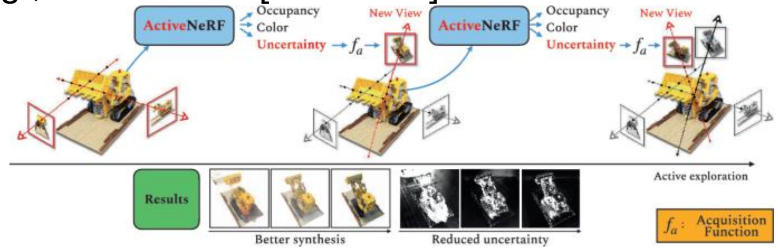


Source: 3DGS [Kerbl 2023]

Novel NeRF+Navigation Architectures

<u>Active sensing / view planning:</u>

- Use NeRF to render unseen parts for improved reasoning, e.g., object detection, task inference
- Find the next-best-view for autonomous exploration, e.g., ActiveNeRF [Pan 2022]



<u>End-to-end navigation with NeRF:</u>

- Unify Localization, Planning, and Control with online NeRF training, e.g., NICE-SLAM [Zhu 2022]
- Integrating multimodal inputs to optimize NeRF-based methods

References

[Mildenhall 2020] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, andR. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. InECCV, 2020.

[Barron 2022] Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P., & Hedman, P. (2022). Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5470-5479).

[Adamkiewicz 2022] M. Adamkiewicz, T. Chen, A. Caccavale, R. Gardner, P. Culbertson, J. Bohg,and M. Schwager. Vision-Only Robot Navigation in a Neural Radiance World.IEEE Robotics and Automation Letters (RA-L), 7(2):4606–4613, Apr. 2022.

[Chen 2024] T. Chen, P. Culbertson, and M. Schwager. Catnips: Collision avoidance throughneural implicit probabilistic scenes. IEEE Transactions on Robotics, 40:2712–2728, Apr. 2024.

[Kerbl 2023] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splattingfor real-time radiance field rendering. ACM Transactions on Graphics, 42(4),July 2023.

[Tancik 2021] M. Tancik, B. Mildenhall, T. Wang, D. Schmidt, P. P. Srinivasan, J. T. Bar-ron, and R. Ng. Learned initializations for optimizing coordinate-based neural representations. In CVPR, 2021.

[Müller 2022] Müller, T., Evans, A., Schied, C., & Keller, A. (2022). Instant neural graphics primitives with a multiresolution hash encoding. ACM transactions on graphics (TOG), 41(4), 1-15.

[Garbin 2021] S. J. Garbin, M. Kowalski, M. Johnson, J. Shotton, and J. Valentin. Fastnerf:High-fidelity neural rendering at 200fps. arXiv preprint arXiv:2103.10380, 2021.

[Pumarola 2021] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer. D-NeRF: NeuralRadiance Fields for Dynamic Scenes. In Proceedings of the IEEE/CVF Confer-ence on Computer Vision and Pattern Recognition, 2021

[Zhu 2022] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Polle-feys. Nice-slam: Neural implicit scalable encoding for slam. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),2022.

Thank you for listening!

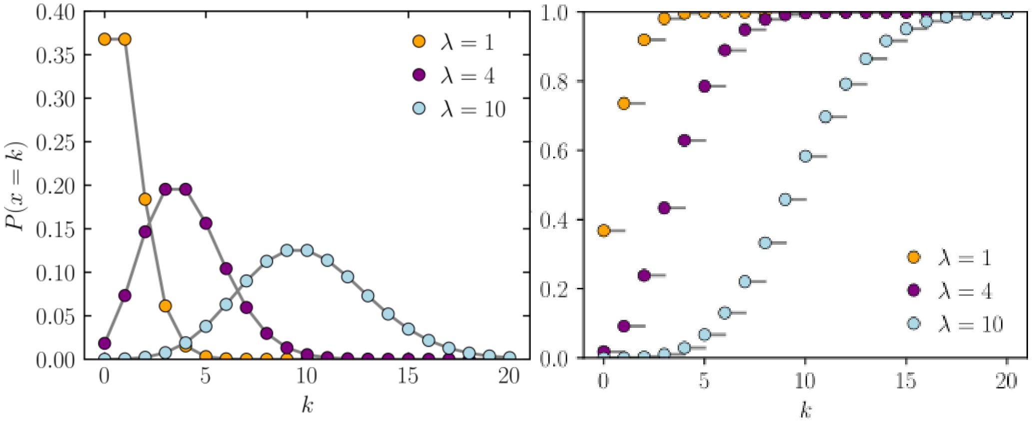
Questions & Comments are welcomed!

Poisson Point Process (PPP)

Poisson Random Variables

First, we recall that a discrete random variable (RV) N that takes values in $\mathbb{N}$ is said to have a Poisson distribution with parameter $\lambda \geq 0$ if its probability mass function is given by

$$Pr(N = m) = \frac{\lambda^m \exp(-\lambda)}{m!}.$$



PPP on multi-dimensional Euclidean space

Definition 1 (Poisson Point Process)**.** *Consider a random process N on $\mathbb{R}^n$ that maps subsets[2] $B \subset \mathbb{R}^n$ of the state space to the random number $N(B)$ of points that lie in B. We say N is a Poisson Point Process (PPP) with intensity $\lambda : \mathbb{R}^n \mapsto \mathbb{R}_+$ if:*

(i) The number of points $N(B)$ that lie in B is a Poisson RV with distribution

$$Pr(N(B) = m) = \frac{\Lambda(B)^m \exp(-\Lambda(B))}{m!},$$

where $\Lambda(B) = \int_{\mathbf{x} \in B} \lambda(\mathbf{x})\, d\mathbf{x}$.

(ii) For k disjoint subsets $B_1, \ldots B_k \subset \mathbb{R}^n$, the number of points in each subset, $N(B_1), \ldots, N(B_k)$, are independent RVs.

bution, the expected number of points in the set B is identical to the integral of the intensity, in other words,

$$\mathbb{E}\left[N(B)\right] = \int_B \lambda(\mathbf{x})\, d\mathbf{x}. \quad (4)$$

Source: CATNIPS [Chen 2024], Wikipedia.org

Collision Prob. under PPP Explanation

$$Pr(N(B(\mathbf{p},\mathbf{R})) \leq N_{\text{aux}}^{\text{max}}; \Lambda_B) = \exp^{-\Lambda_B} \sum_{i=0}^{\lfloor N_{\text{aux}}^{\text{max}} \rfloor} \frac{\Lambda_B^i}{i!} \geq \sigma \tag{4}$$

$B(\mathbf{p},\mathbf{R})$: robot pose,

$N(B(\mathbf{p},\mathbf{R}))$: actual number of particles in the robot volume at the current pose, calculated by PPP CDF and intensity Λ_B,

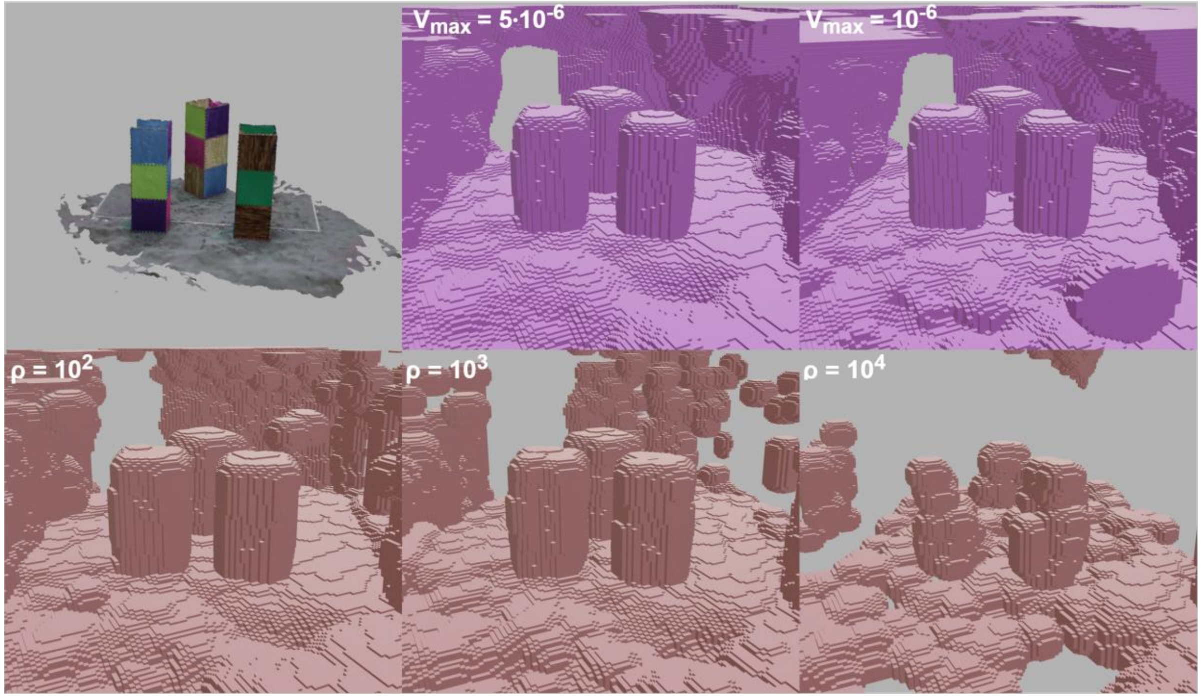
$N_{\text{aux}}^{\text{max}}$: maximum number of particles that should exist in the robot volume,

Λ_B: intensity for the particles over the robot body volume,

$\sigma = 95\%, 99\%$: desired probability threshold.

Source: CATNIPS [Chen 2024], Wikipedia.org

Mesh, PURR, Basegrid



Source: CATNIPS [Chen 2024]