HSE UNIVERSITY

p(B|A)yesgroup.ru

Reconstruction of 3D Scenes

An Overview of Neural Fields for Novel View Synthesis

Kirill Struminsky
Research Fellow @ HSE Centre for Deep Learning and Bayesian Methods

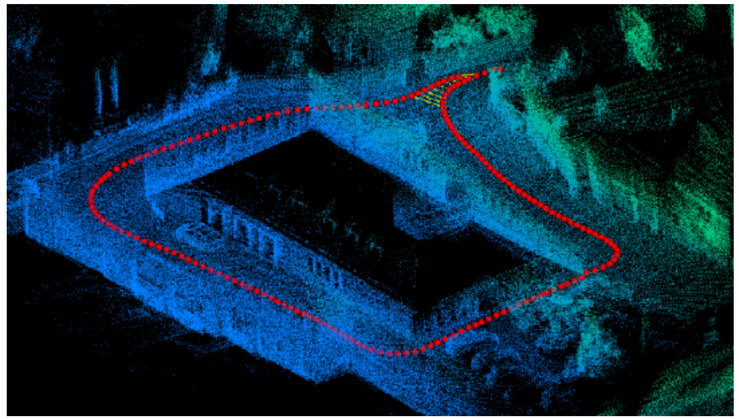
Agenda

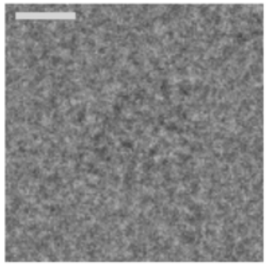
- Computer graphics basics
- Neural fields for representing low-dimensional data
- Novel view synthesis

How to Represent and Reconstruct the World Around Us?

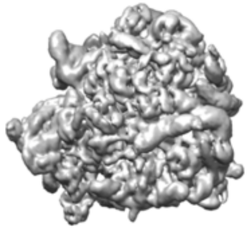
Data examples

- Navigation in robotics
  - Mono or stereo cameras
  - Depth sensors, e.g. LiDAR
- Visualisation
  - MRI
  - Electron Microscopy
  - Video conferencing





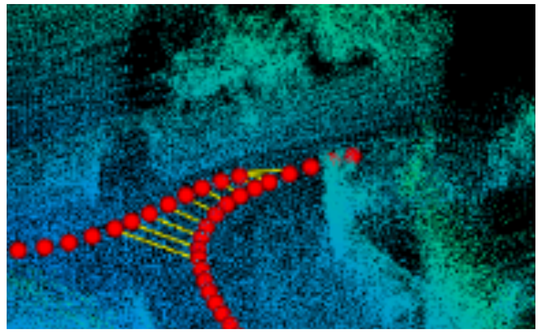
$10^{4\text{-}7}$ projection images



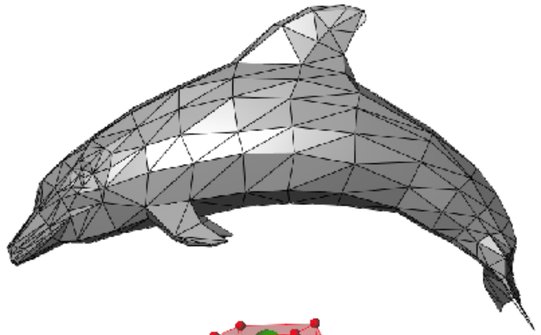
3D electron density

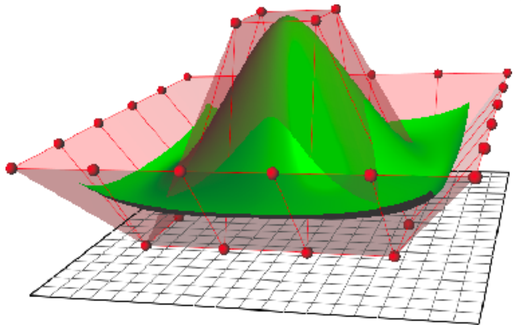
- https://www.ifp.uni-stuttgart.de/en/research/photogrammetric_computer_vision/SLAM/
- https://www.tesla.com/tesla-gallery
- https://ru.wikipedia.org/wiki/Беспилотные_автомобили_Яндекса
- Marc Levoy, Efficient Ray Tracing of Volume Data
- Ellen Zhong et al., Reconstructing continuous distributions of 3D protein structure from cryo-EM images
- Project Starline: Feel like you're there, together, https://www.youtube.com/watch?v=Q13CishCKXY

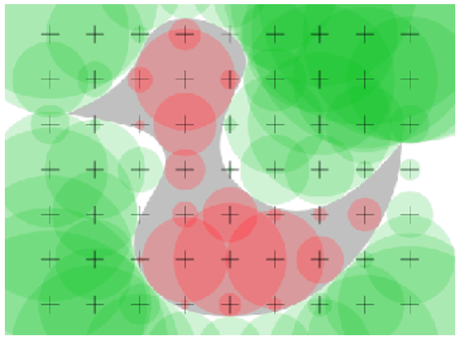
Common Representations



- Explicit
  - Point clouds
  - Polygonal meshes
  - Parametric $x = g(t)$
- Implicit
  - Isosurfaces $x : f(x) = 0$







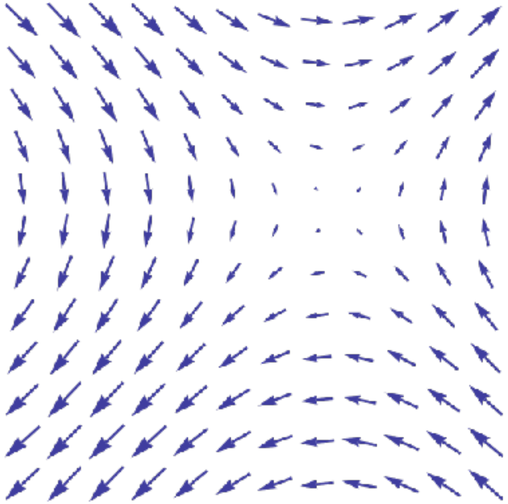
Volumetric Representations

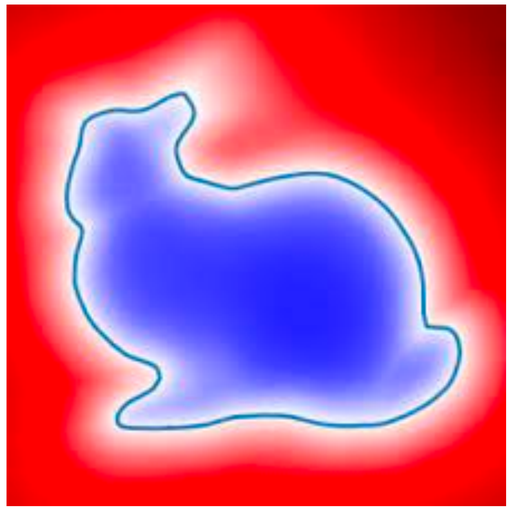
- Surface is a common abstraction in computer graphics
  - Convenience, hardware optimization
  - Sometimes fail to represent real world phenomena
- As an alternative, we will represent scenes with “density” functions
  - Is there something in this point of space?
  - Does light pass through this point in space?
- Scalar **field:** each spatial point stores a scalar value

https://www.reddit.com/r/MicrosoftFlightSim/comments/ymz0ex/is_this_really_what_photogrammetry_looks_like_for/

Vector Fields

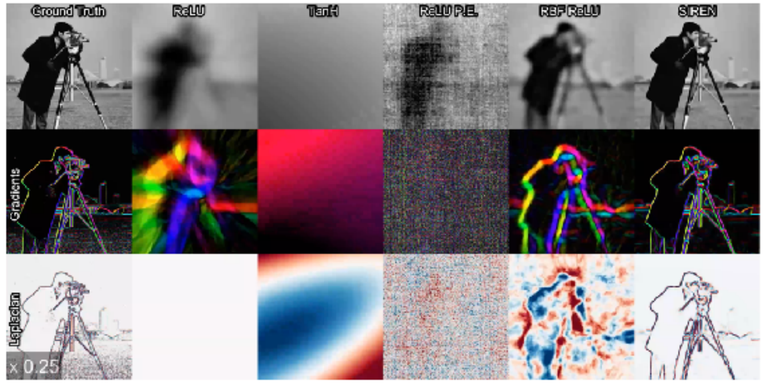
- Used in physics and beyond
- Relevant examples
  - Light absorption coefficients
  - Signed Distance Fields
  - Image

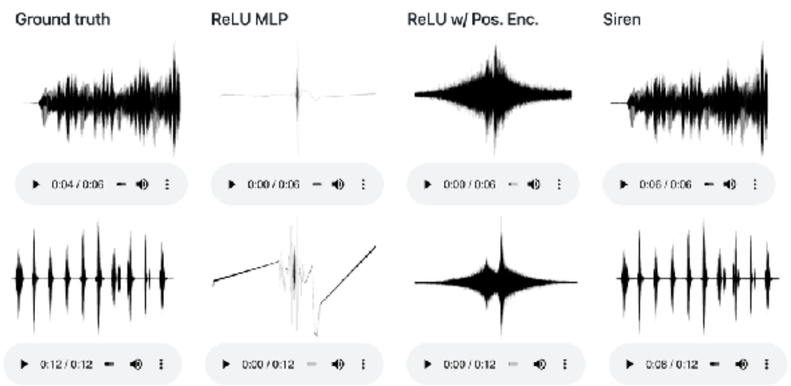




https://en.wikipedia.org/wiki/X-ray
https://arxiv.org/abs/1901.05103
https://en.wikipedia.org/wiki/Lenna

Neural Fields





Input: a point in space

Output: vector

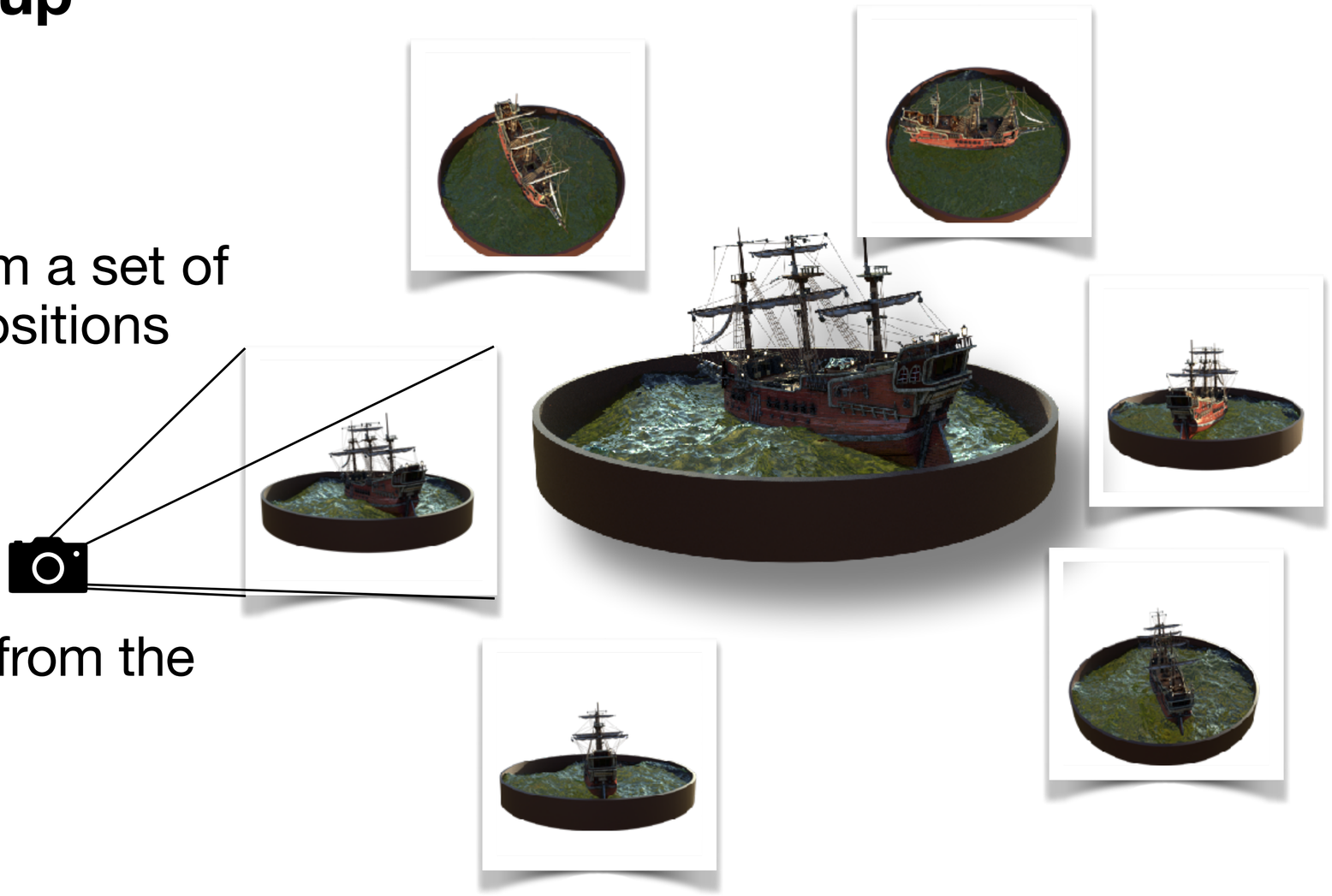
- Architecture: MLP (+ positional embeddings)
- Use cases
  - Photo
  - Video
  - Audio

Sitzmann V. et al. Implicit neural representations with periodic activation functions //Advances in Neural Information Processing Systems. – 2020. – Т. 33. – С. 7462-7473.

Neural Radiance Fields

Novel View Synthesis Setup

- Problem setup:
  - A set of pictures taken from a set of pre-determined camera positions
  - Test camera position
- Goal:
  - A picture of a scene taken from the test camera position



Mildenhall B. et al. Nerf: Representing scenes as neural radiance fields for view synthesis //Communications of the ACM. – 2021. – Т. 65. – №. 1. – С. 99-106.

Neural Radiance Fields

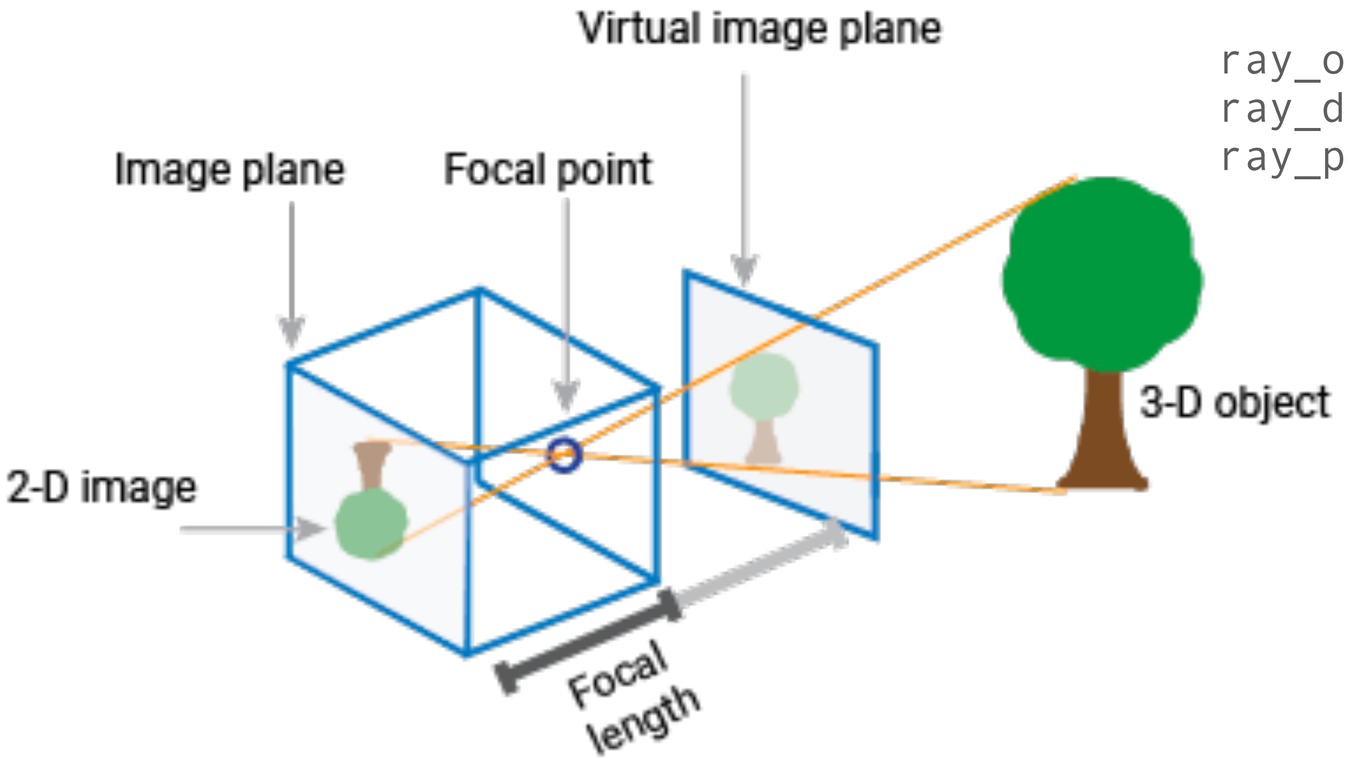
Representing a Scene with Neural Fields

- Represent a scene with two fields
  - Density: $\sigma(x) : \mathbb{R}^3 \rightarrow \mathbb{R}^+$
  - Radiance: $C(x, d) : \mathbb{R}^3 \times S^2 \rightarrow \mathbb{R}^3$
- Density represents is related to opacity
- Radiance represents color of a point
- Architecture: MLP + positional embeddings

Mildenhall B. et al. Nerf: Representing scenes as neural radiance fields for view synthesis //Communications of the ACM. – 2021. – Т. 65. – №. 1. – С. 99-106.

How an Image is Formed?

Camera Model



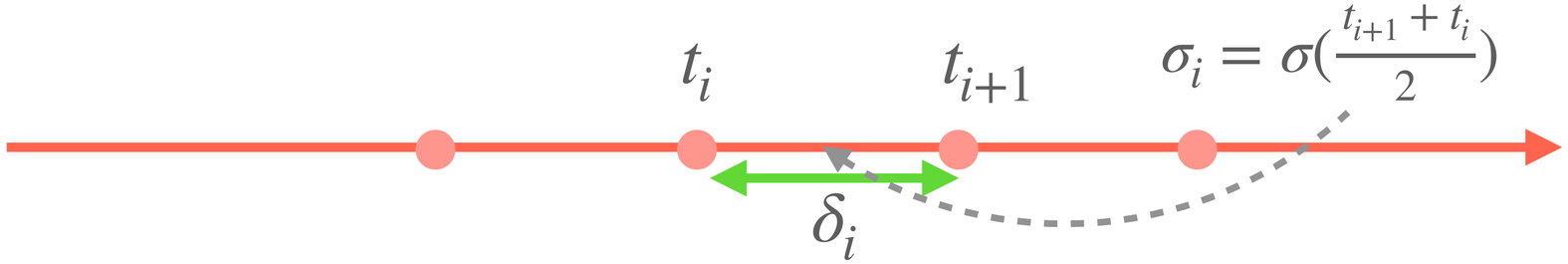

```
Generate a point on a ray:
(x, y) \in [-1, 1]^2 - pixel coordinates
f - focal length
t - distance between the camera and the ray point

ray_origin = (0, 0, 0)
ray_direction = normalize((x, y, f) - ray_origin)
ray_point = ray_origin + t * ray_direction
```

https://www.mathworks.com/help/vision/ug/camera-calibration.html

Computing Pixel Color

- Density $\sigma(t) \in [0, +\infty)$ is related to opacity at t
- Divide the ray with points $t_1, \ldots, t_n$ and define

$$\alpha_i = 1 - \exp(-\sigma_i \delta_i)$$



- Expected color along a ray is given by

$$C = \sum_i C(t_i) \cdot \alpha_i \prod_{j<i} (1 - \alpha_j)$$

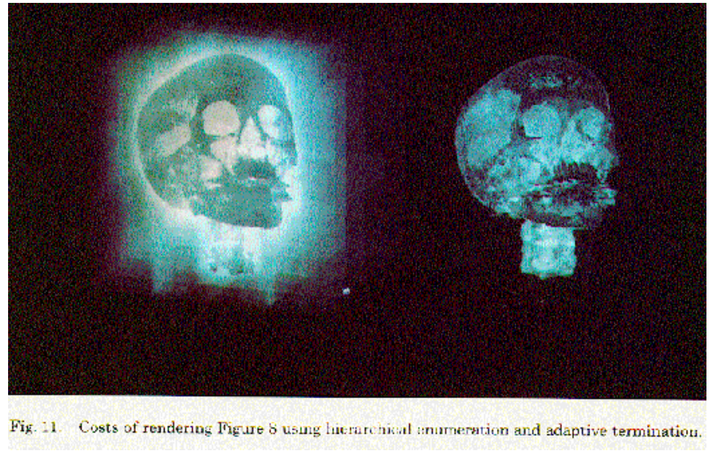


Fig. 11. Costs of rendering Figure 8 using hierarchical enumeration and adaptive termination.

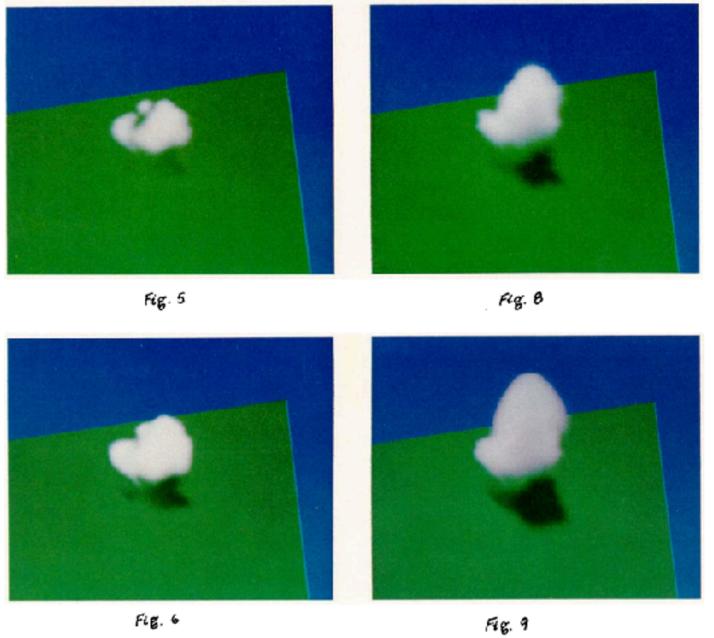


Fig. 5 Fig. 8

Fig. 6 Fig. 9

Neural Radiance Fields

Optimisation

- Training: minimise the mean squared error $\mathbb{E}_D \| C - C_{gt} \|^2$

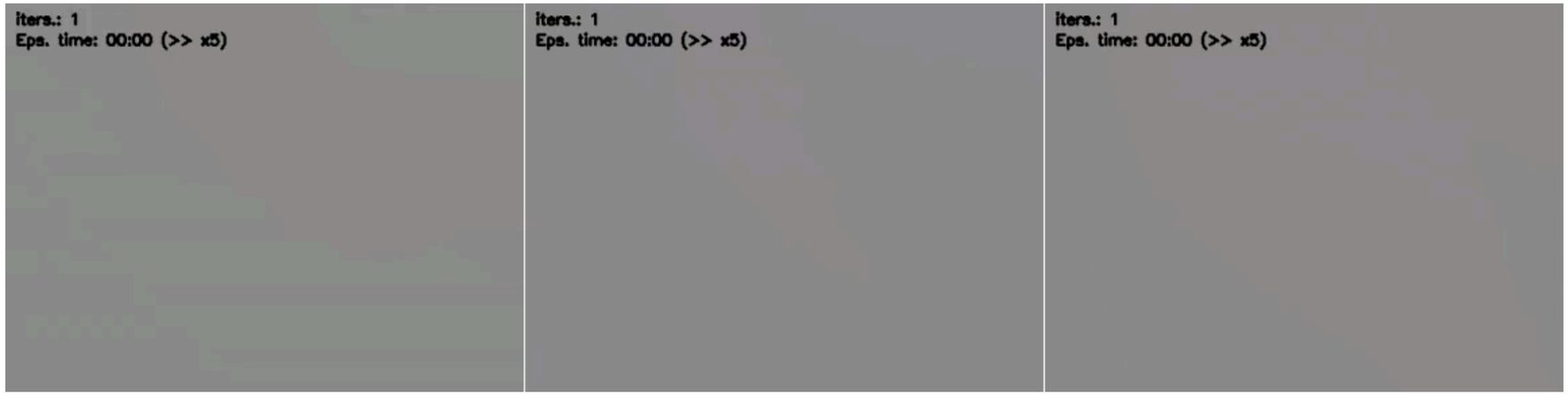
$$C = \sum_i C(x_i, d) \alpha_i \prod_{j<i} (1 - \alpha_j)$$

  - Optimise w.r.t. parameters of radiance C and density σ

Training Visualisation

- Synthetic data

- Real forward-facing

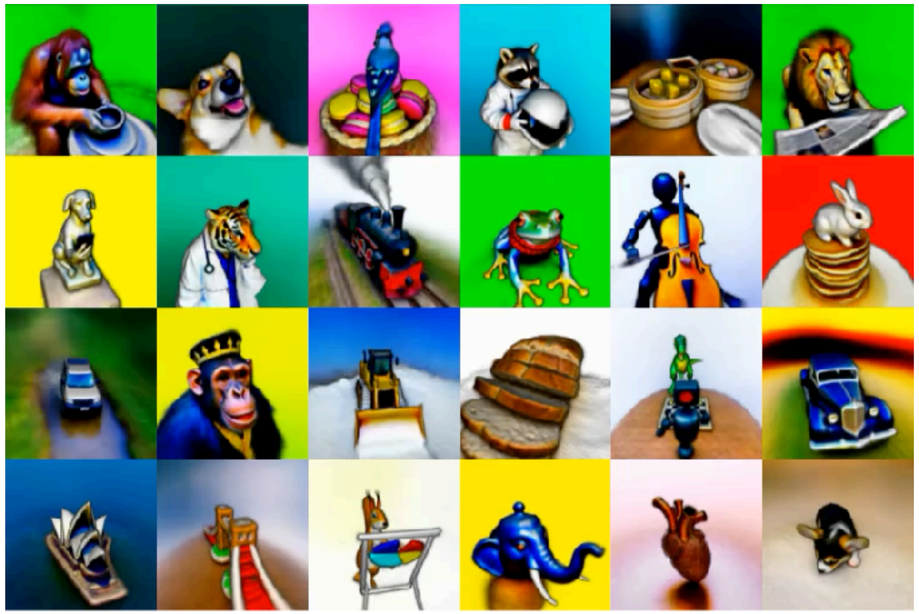


https://github.com/sunset1995/DirectVoxGO

Illustrating Model Flexibility

Multimodal Data

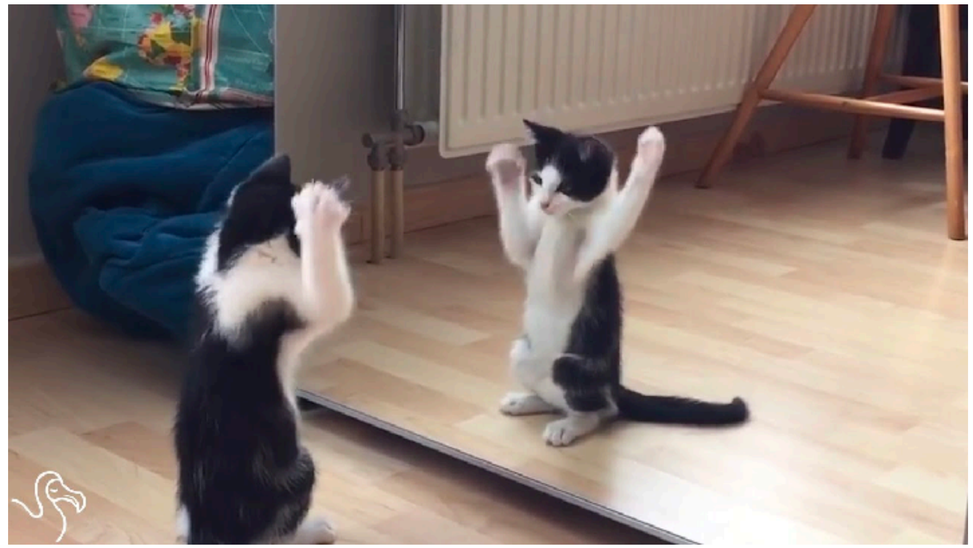
- One can replace MSE reconstruction loss with fancier training signals
  - Leverage text-to-2D diffusion model for scene generation *(second lecture)*
  - CLIP-embeddings for text retrieval and 3D segmentation



Poole B. et al. Dreamfusion: Text-to-3d using 2d diffusion //arXiv preprint arXiv:2209.14988. – 2022.
Kerr J. et al. LERF: Language Embedded Radiance Fields //arXiv preprint arXiv:2303.09553. – 2023.

NeRF pros and cons

+ Simplicity & Flexibility
+ Photorealism
+ Compression *(5mb / scene)*
- Ill-posed problem
- Does not work on real scenes
- Long training time *(48 h. / scene)*
- Slow rendering *(1 min. / frame)*



Improving Rendering Speed

Sparsification

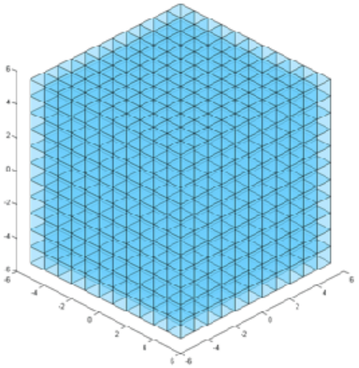
- Each terms in the sum requires running an MLP

$$C = \sum_i C(x_i)\alpha_i \prod_{j<i} (1 - \alpha_j)$$

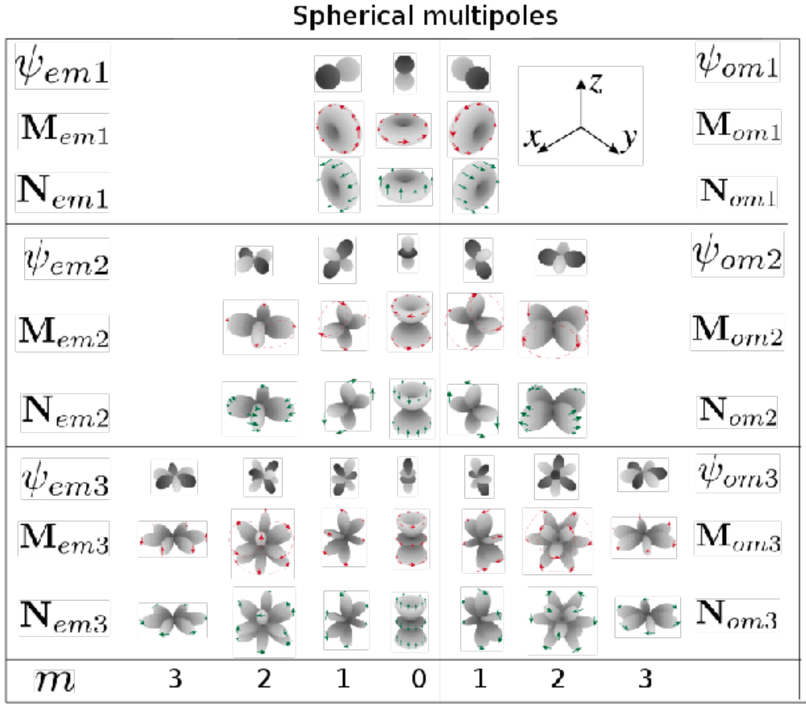
- If $\sigma(x_i) = 0$, then $1 - \alpha_i = 0$, we can omit i-th term
- Idea: cache $\sigma(\cdot)$, omit i if we know $\sigma(x_i) = 0$
- Outcome: training and rendering 10-30 times faster

Liu L. et al. Neural sparse voxel fields //Advances in Neural Information Processing Systems. – 2020. – T. 33. – C. 15651-15663.
Hedman P. et al. Baking neural radiance fields for real-time view synthesis //Proceedings of the IEEE/CVF International Conference on Computer Vision. – 2021. – C. 5875-5884.
Sun C., Sun M., Chen H. T. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2022. – C. 5459-5469.
Fridovich-Keil S. et al. Plenoxels: Radiance fields without neural networks //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2022. – C. 5501-5510.
Li R., Tancik M., Kanazawa A. NerfAcc: A General NeRF Acceleration Toolbox //arXiv preprint arXiv:2210.04847. – 2022.

Volume Grid Representations



- There is no magic in using neural nets
  - We can represent $\sigma(x) : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ with a volumetric grid
  - Radiance $C(x, d) : \mathbb{R}^3 \times S^2 \rightarrow \mathbb{R}^3$ involves 5D grid
- Spherical harmonics - basis for functions on spheres
- Each spatial point x stores basis coordinates $C(x, \cdot)$
- Training time: minutes
- Real-time rendering



Yu A. et al. Plenoctrees for real-time rendering of neural radiance fields //Proceedings of the IEEE/CVF International Conference on Computer Vision. – 2021. – С. 5752-5761.
Yu A. et al. Plenoxels: Radiance fields without neural networks //arXiv preprint arXiv:2112.05131. – 2021.

Hybrid Approach

- There is no magic in using neural nets
  - We can represent $\sigma(x) : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ with a volumetric grid
  - Radiance $C(x, d) : \mathbb{R}^3 \times S^2 \rightarrow \mathbb{R}^3$ involves 5D grid
- Hybrid representation $C(x, d) = F(G(x), d)$:
  - $G(x)$ - vector field, represented with a voxel grid
  - $F(v, d)$ - tiny MLP
- Leads to higher fidelity compared to spherical harmonics

Sun C., Sun M., Chen H. T. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2022. – С. 5459-5469.

Instant-NGP

Pros and cons of the above solutions

Is there a solution compromising between the two?

MLP

- Size: ~5mb
- Slow
- Trade-of speed for fidelity

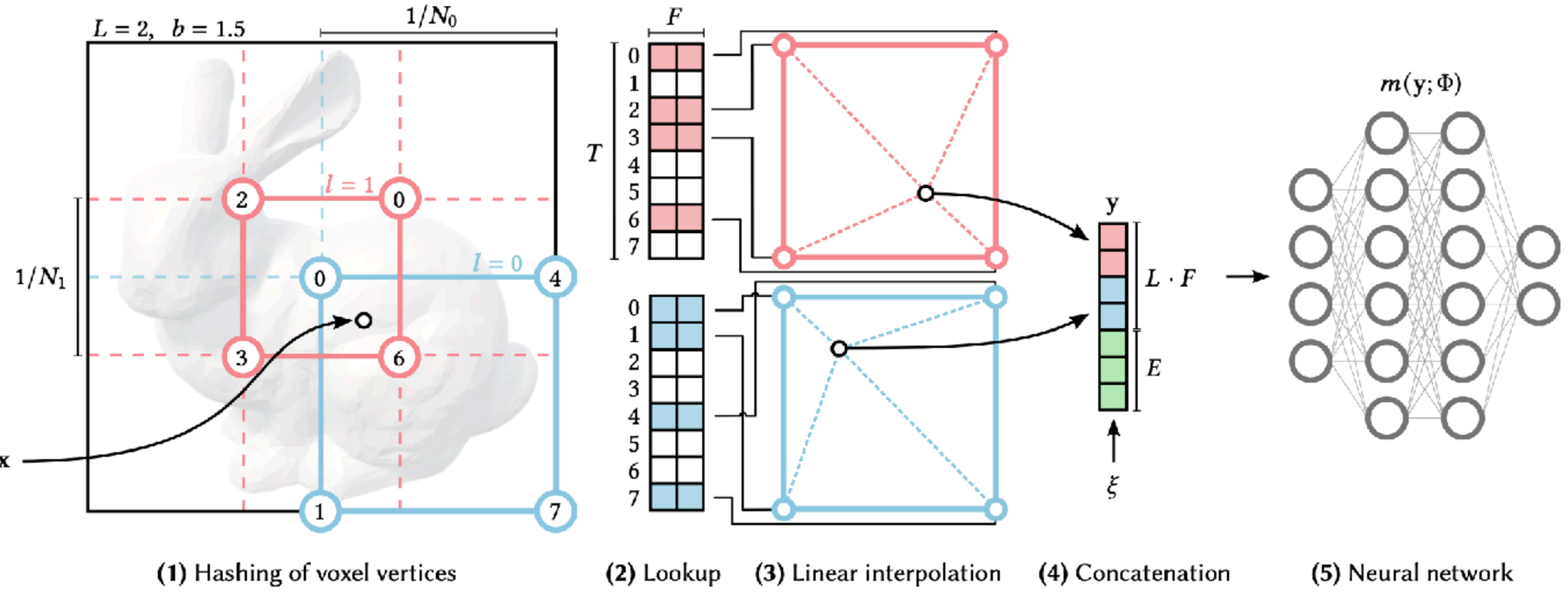
Voxel Grid

- Size: ~1Gb
- Fast
- Trade-off memory for fidelity

Müller T. et al. Instant neural graphics primitives with a multiresolution hash encoding //ACM Transactions on Graphics (ToG). – 2022. – Т. 41. – №. 4. – С. 1-15.

Instant-NGP

Saving time and memory with HashGrids



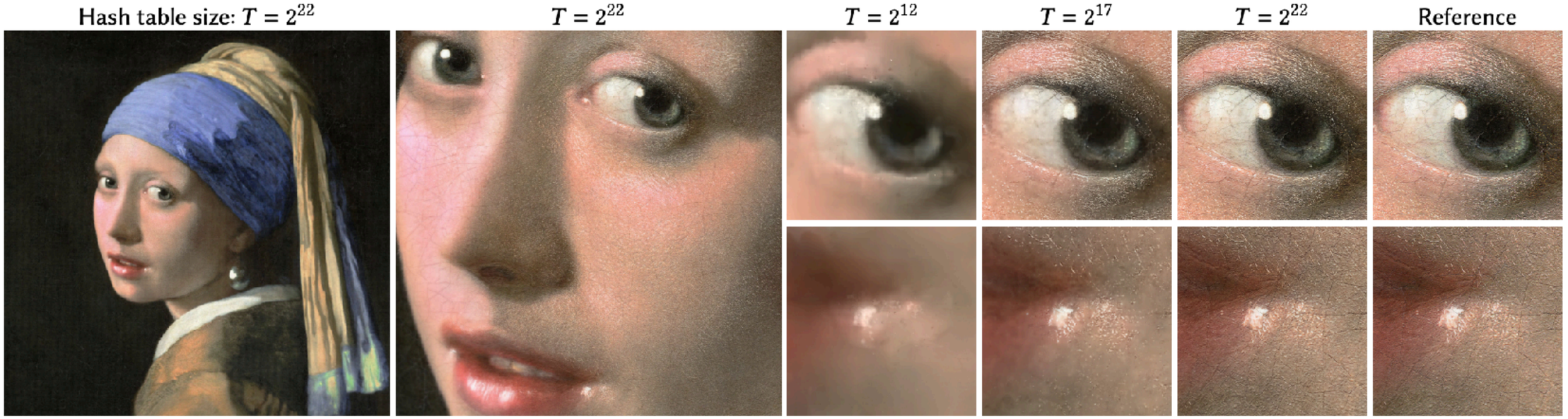
HashGrid

- Size ~100mb
- Average speed
- Flexibility factors
  - HashGrid size
  - MLP component

Müller T. et al. Instant neural graphics primitives with a multiresolution hash encoding //ACM Transactions on Graphics (ToG). – 2022. – Т. 41. – №. 4. – С. 1-15.
https://github.com/NVlabs/instant-ngp

Neural Graphics Primitives

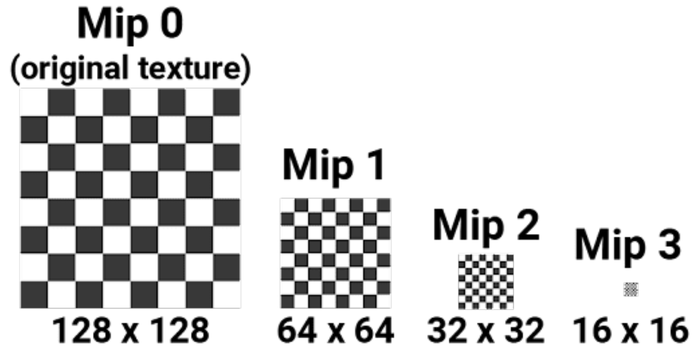
- Network size is 3.4% of the full image size (4×10^8 pixels)
- PSNR 29.8



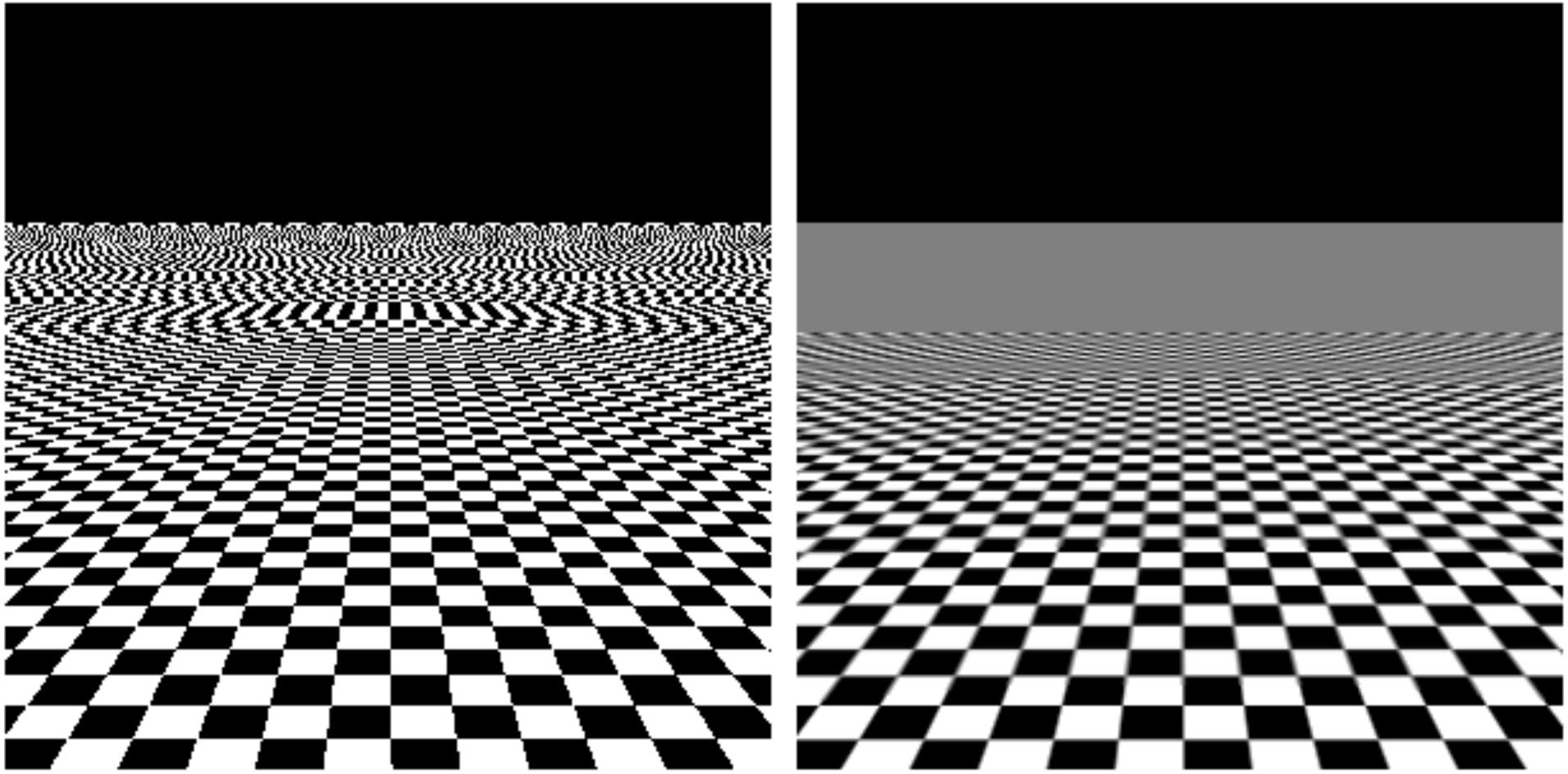
Müller T. et al. Instant neural graphics primitives with a multiresolution hash encoding //ACM Transactions on Graphics (ToG). – 2022. – Т. 41. – №. 4. – С. 1-15.

Improving Reconstruction Fidelity

Aliasing

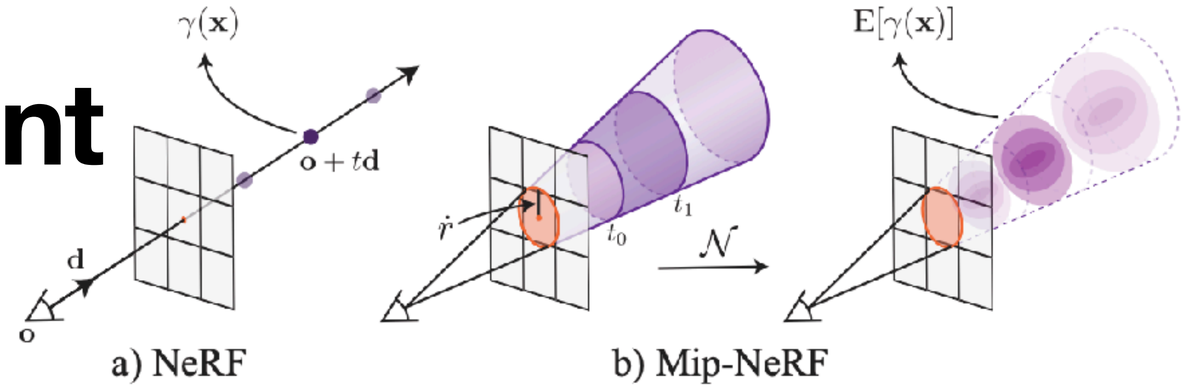


multum in parvo



Taking Scale into Account

A Parallel Line of Reseach

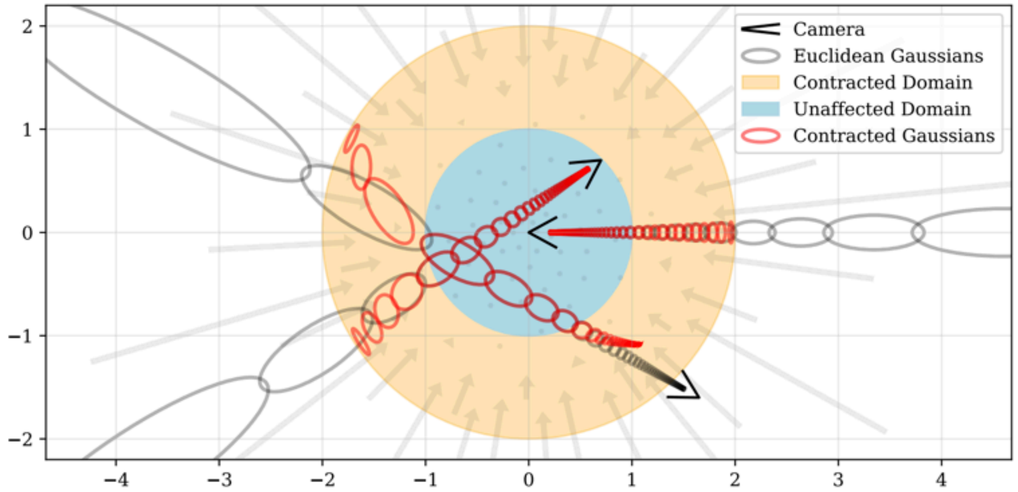


- Original NeRF experiments had biased data
  - Camera distances did not change by a significant amount
- Pixels cover a cone-like region in space
- Ideal solution - compute the average radiance over the whole cone
- Mip-NeRF approximates the average radiance/density over a cone segment
- Idea: compute embedding average rather than output average

Barron J. T. et al. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields //Proceedings of the IEEE/CVF International Conference on Computer Vision. – 2021. – С. 5855-5864.

Unbounded Scenes

- Original NeRF experiments had biased data
  - All the training scenes were bounded
- NeRF++ models foreground and background with two networks
- Mip-NeRF-360 maps the space into a ball
  - Does not transform scene center
  - The rest is mapped onto a shell



Zhang K. et al. Nerf++: Analyzing and improving neural radiance fields //arXiv preprint arXiv:2010.07492. – 2020.
Barron J. T. et al. Mip-nerf 360: Unbounded anti-aliased neural radiance fields //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2022. – С. 5470-5479.

Zip-NeRF

- Merges two research branches
  - Speed: Instant-NGP
  - Fidelity: Mip-NeRF-360
- Mip-NeRF is designed for MLPs
- Zip-NeRF adapts Mip-maps to Instant-NGP

Barron J. T. et al. Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields //arXiv preprint arXiv:2304.06706. – 2023.
https://jonbarron.info/zipnerf/

Further References

- Useful tools:
  - Open source solution for training NeRFs https://docs.nerf.studio/
  - “There’s an app for that” https://lumalabs.ai/
- Hot topics beyond the scope of lecture:
  - NeRFs for video https://dynibar.github.io/
  - Better surface reconstruction: https://research.nvidia.com/labs/dir/neuralangelo/
  - Gaussian Splatting: recent NeRF alternative https://gsplat.tech/