

Нейросетевые поля

для реконструкции сцен и не только

Струминский Кирилл, 16 октября 2023

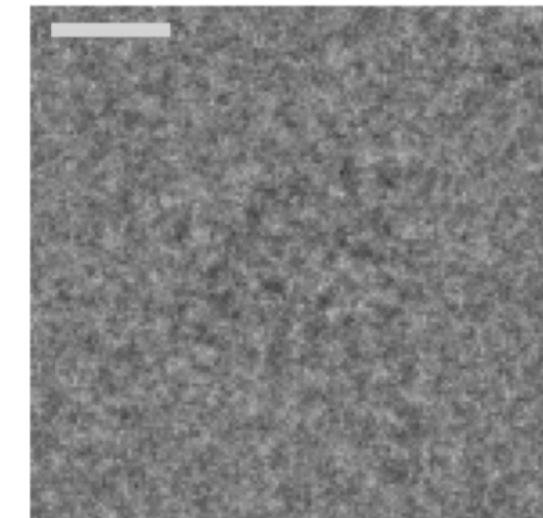
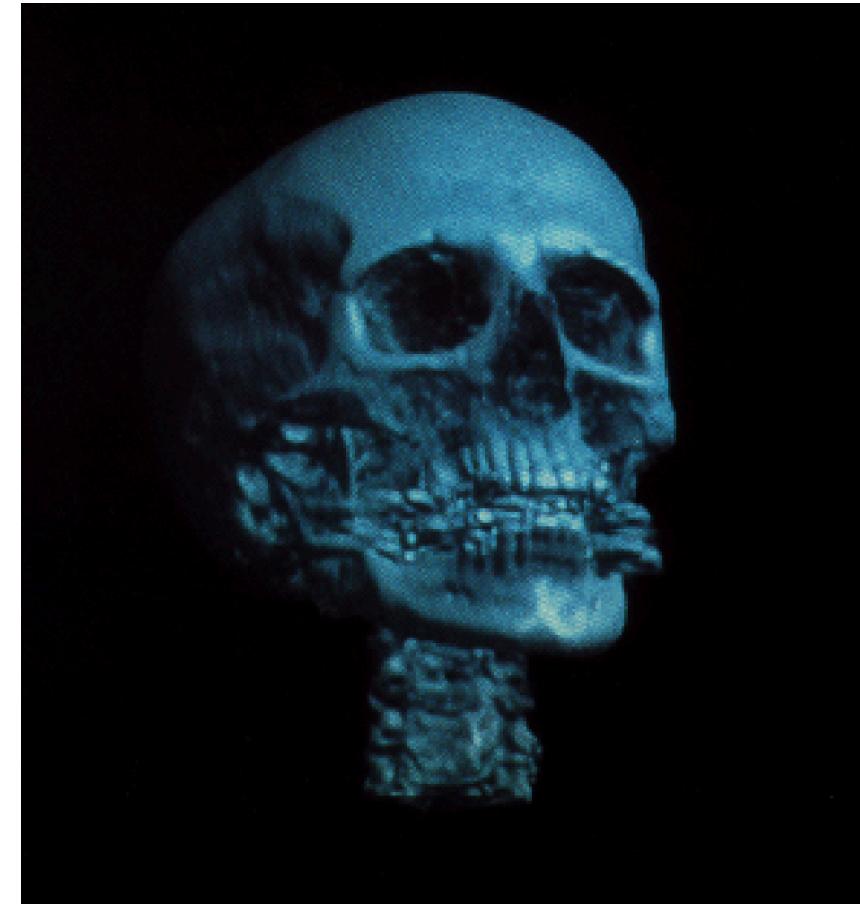
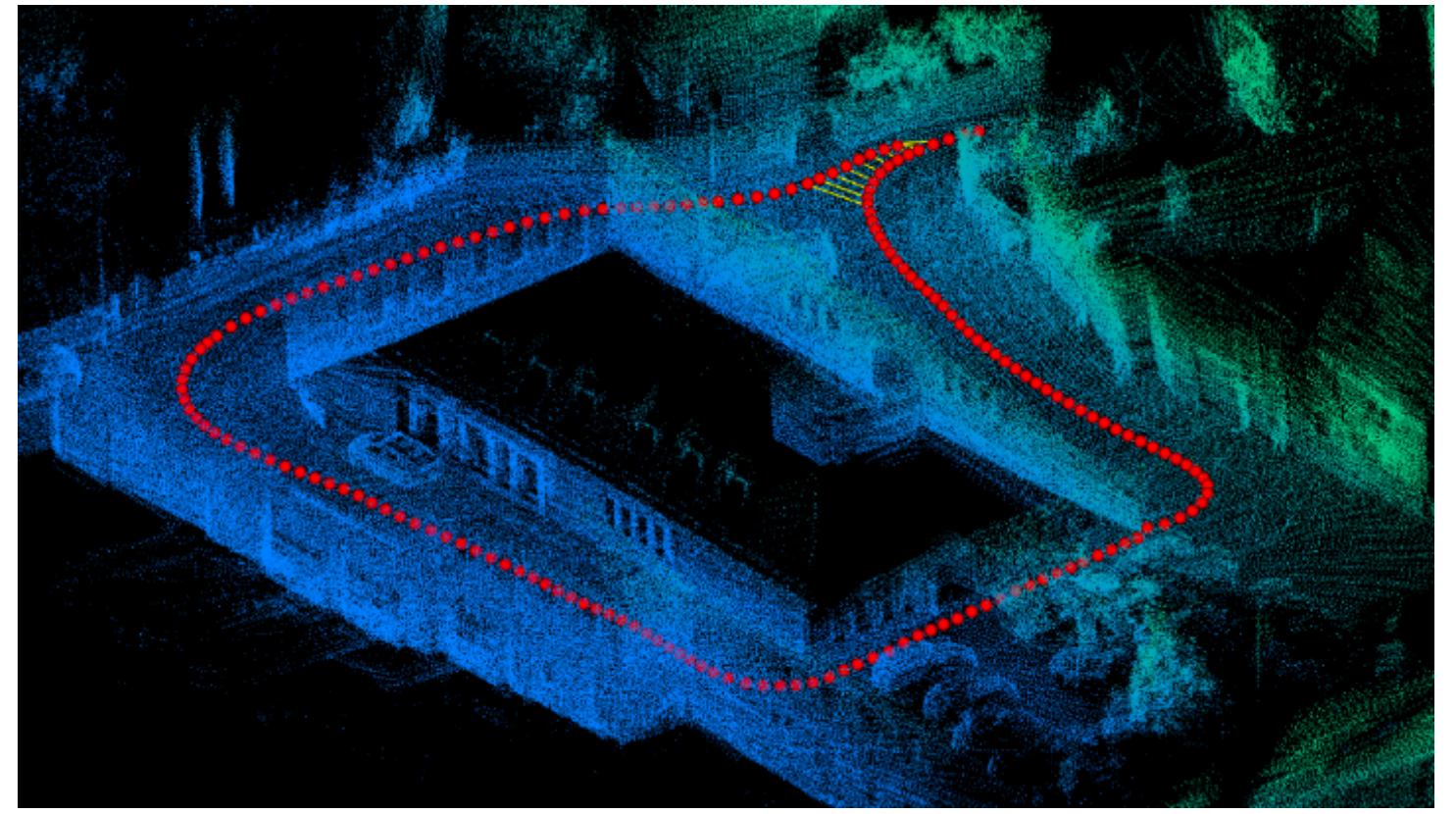
О чём пойдет речь

- О нейросетевых полях: модель Neural Radiance Fields
- О трёхмерных данных в зрении и машинном обучении
- О компьютерной графике

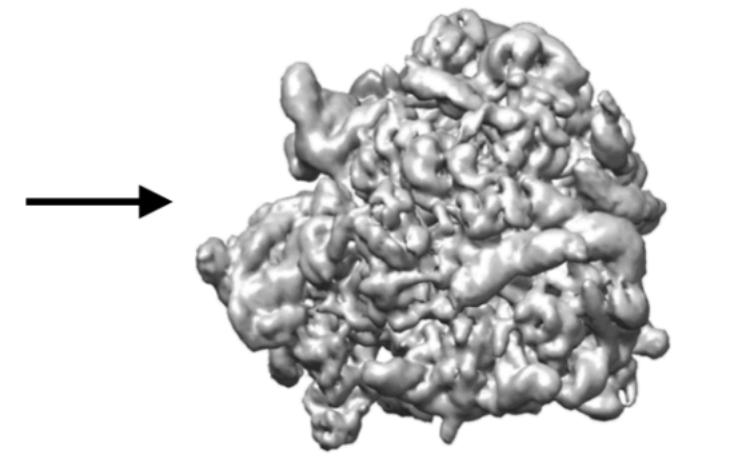
Трёхмерная реконструкция

Многообразие задач

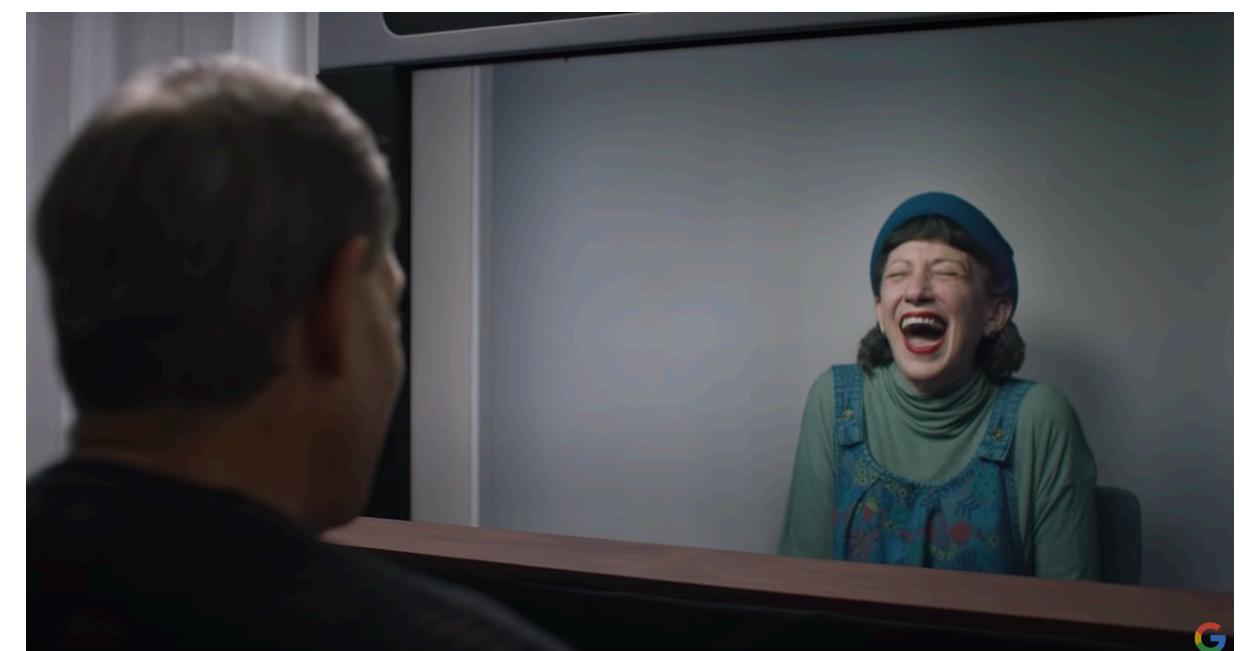
- Навигация в пространстве
 - Одна или несколько камер
 - Камеры и датчики глубины
 - Визуализация
 - Томография
 - Электронная микроскопия
 - Повседневная жизнь



10^{4-7} projection images

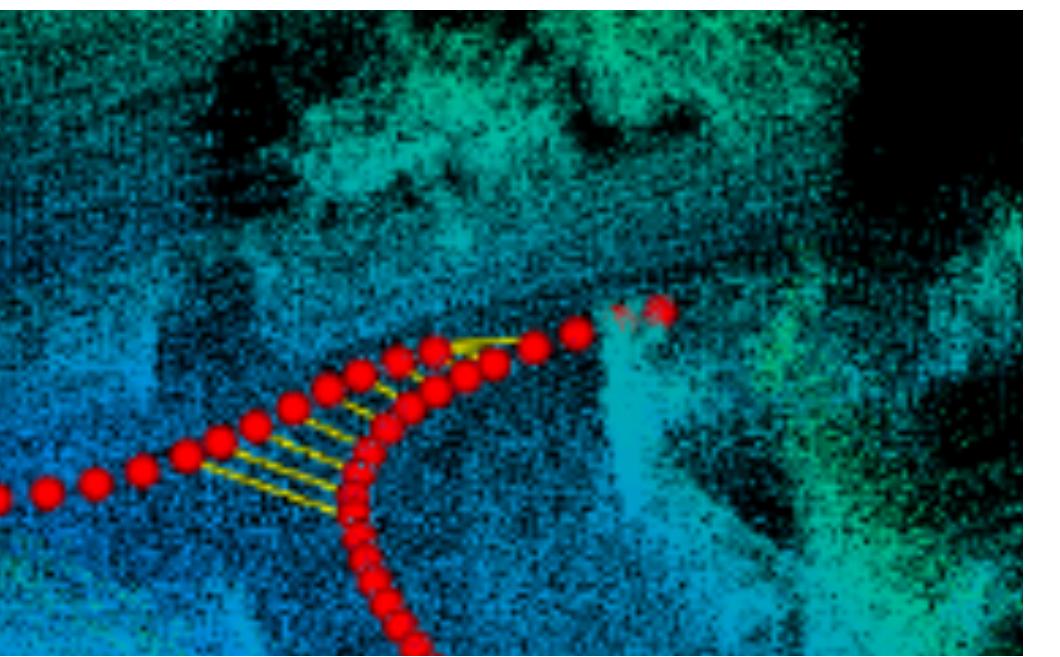


3D electron density

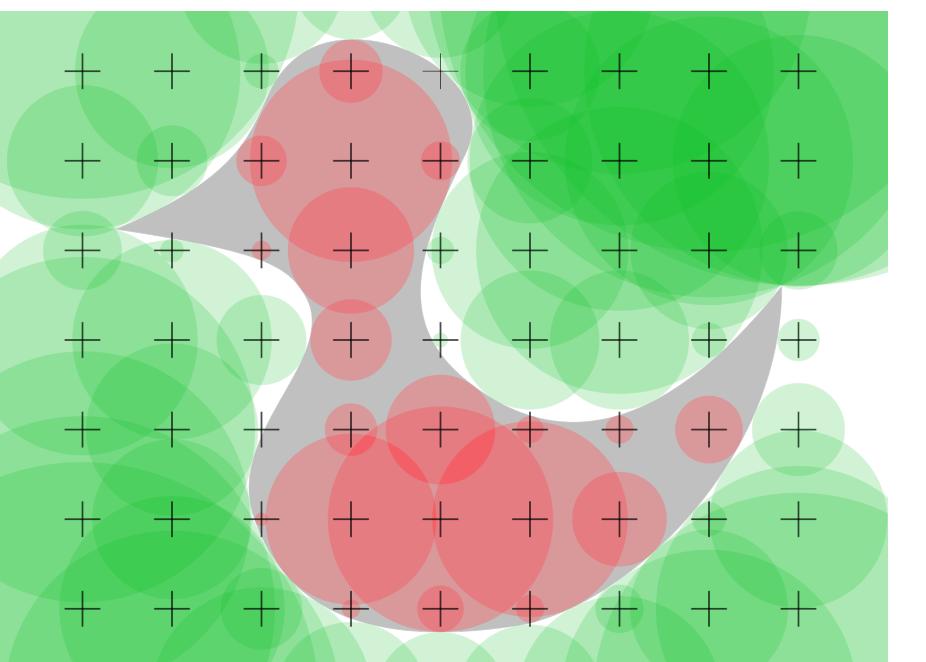
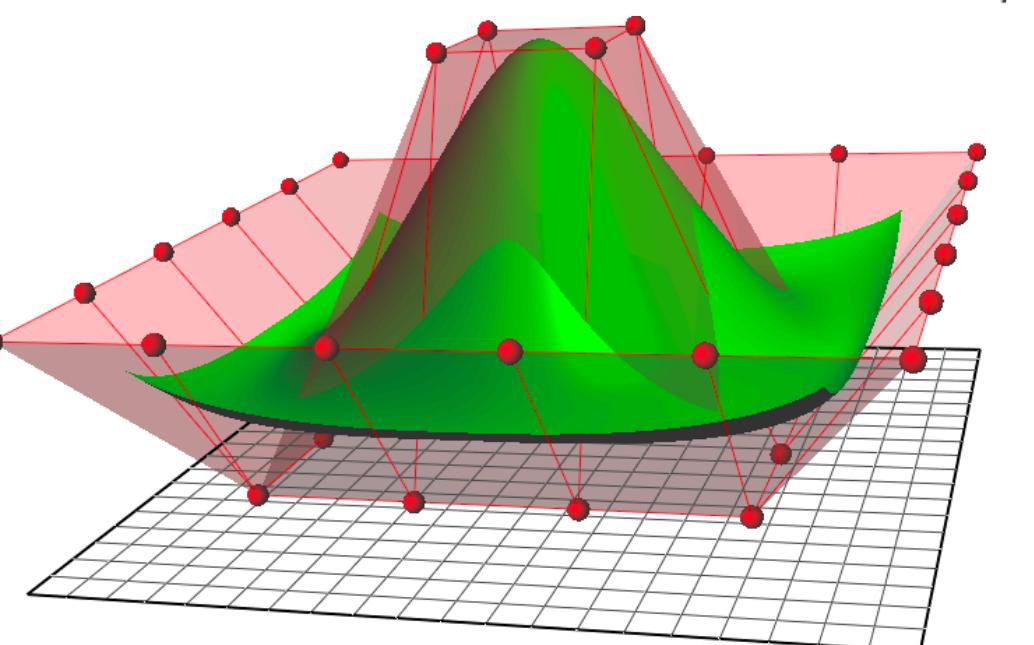
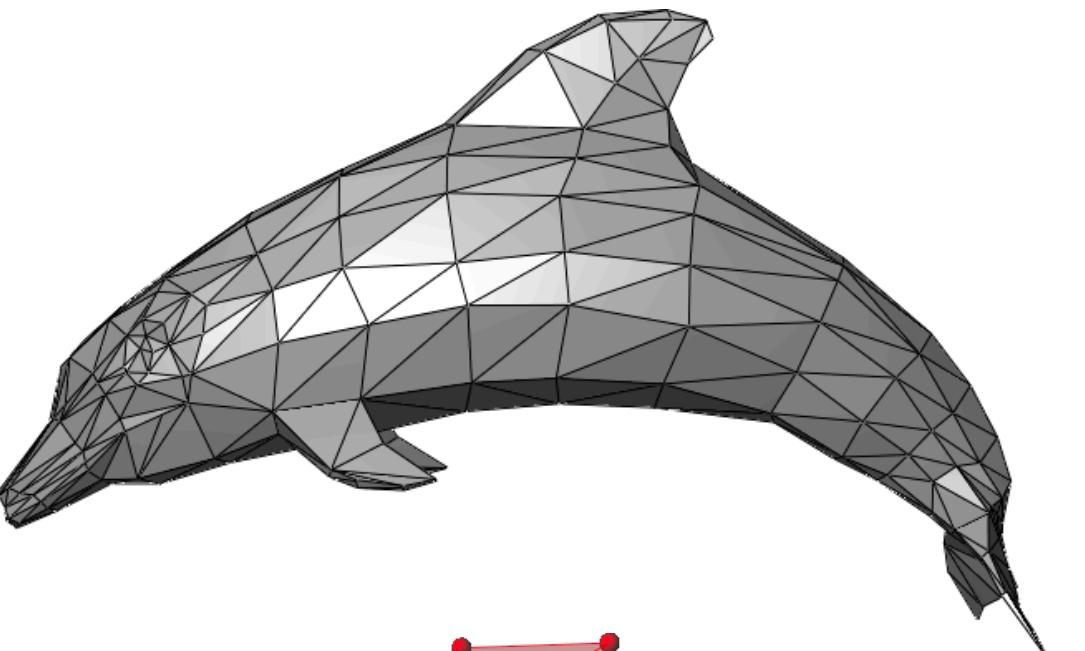


- https://www.ifp.uni-stuttgart.de/en/research/photogrammetric_computer_vision/SLAM/
- <https://www.tesla.com/tesla-gallery>
- https://ru.wikipedia.org/wiki/Беспилотные_автомобили_Яндекса
- Marc Levoy, Efficient Ray Tracing of Volume Data
- Ellen Zhong et al., Reconstructing continuous distributions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

Трёхмерные модели В компьютерной графике



- Явные
 - Облака точек
 - Полигональные сетки
 - Параметрические $x = g(t)$
- Неявные поверхности $x : f(x) = 0$



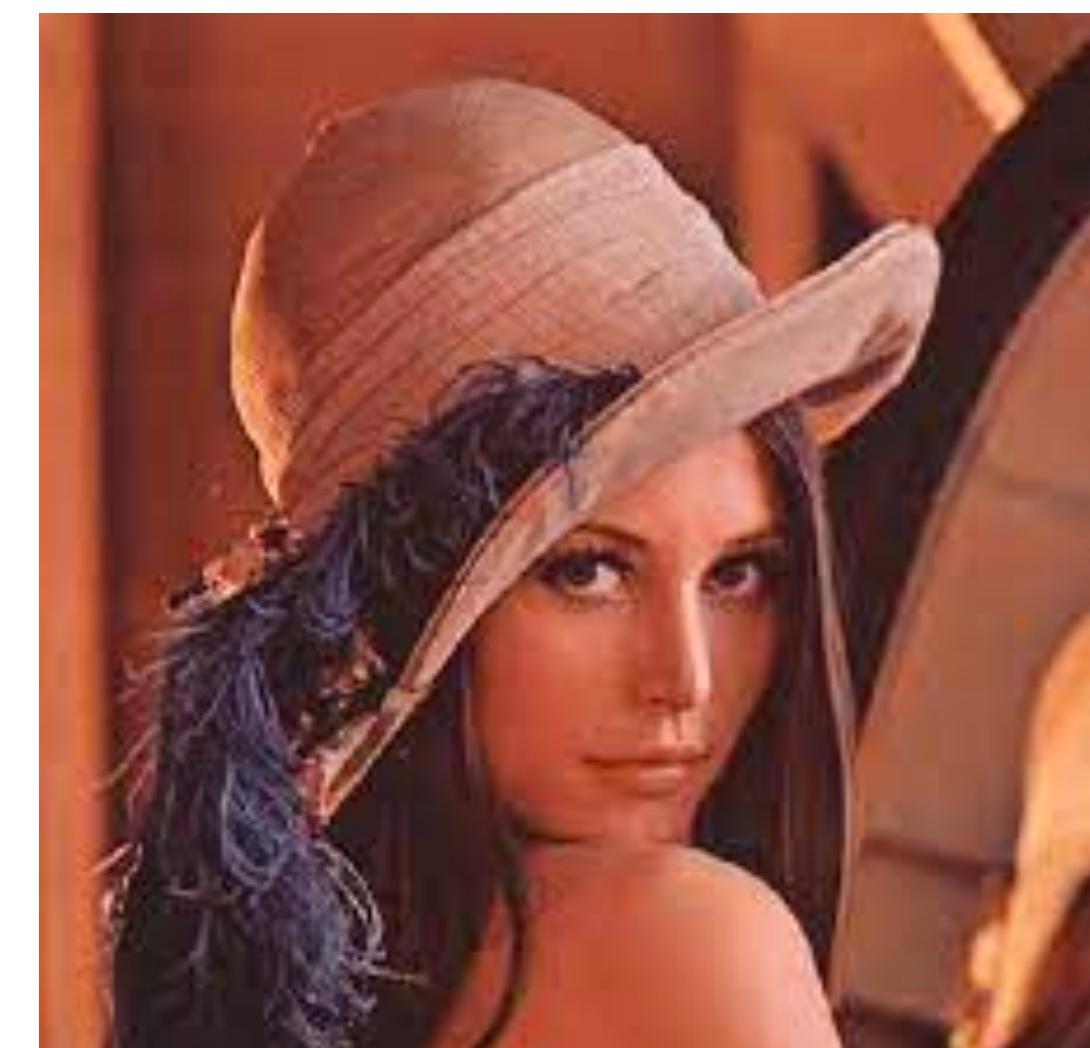
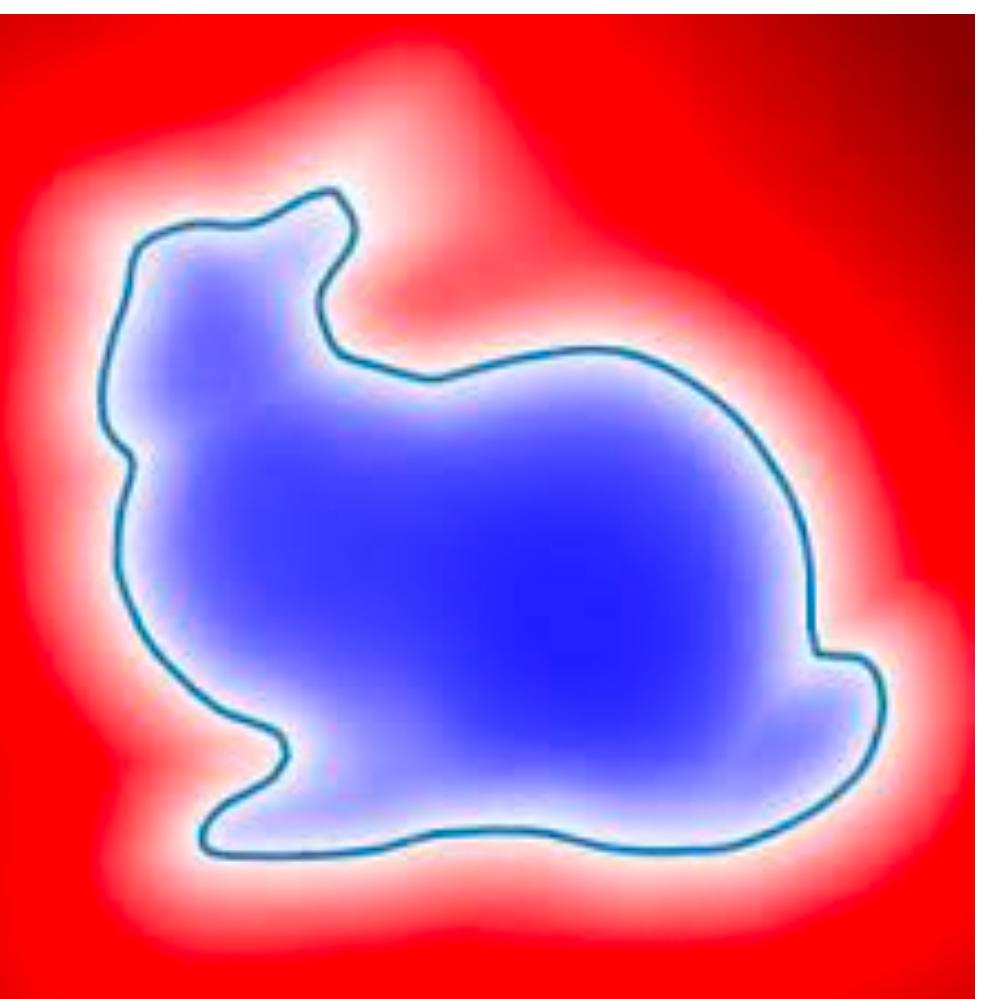
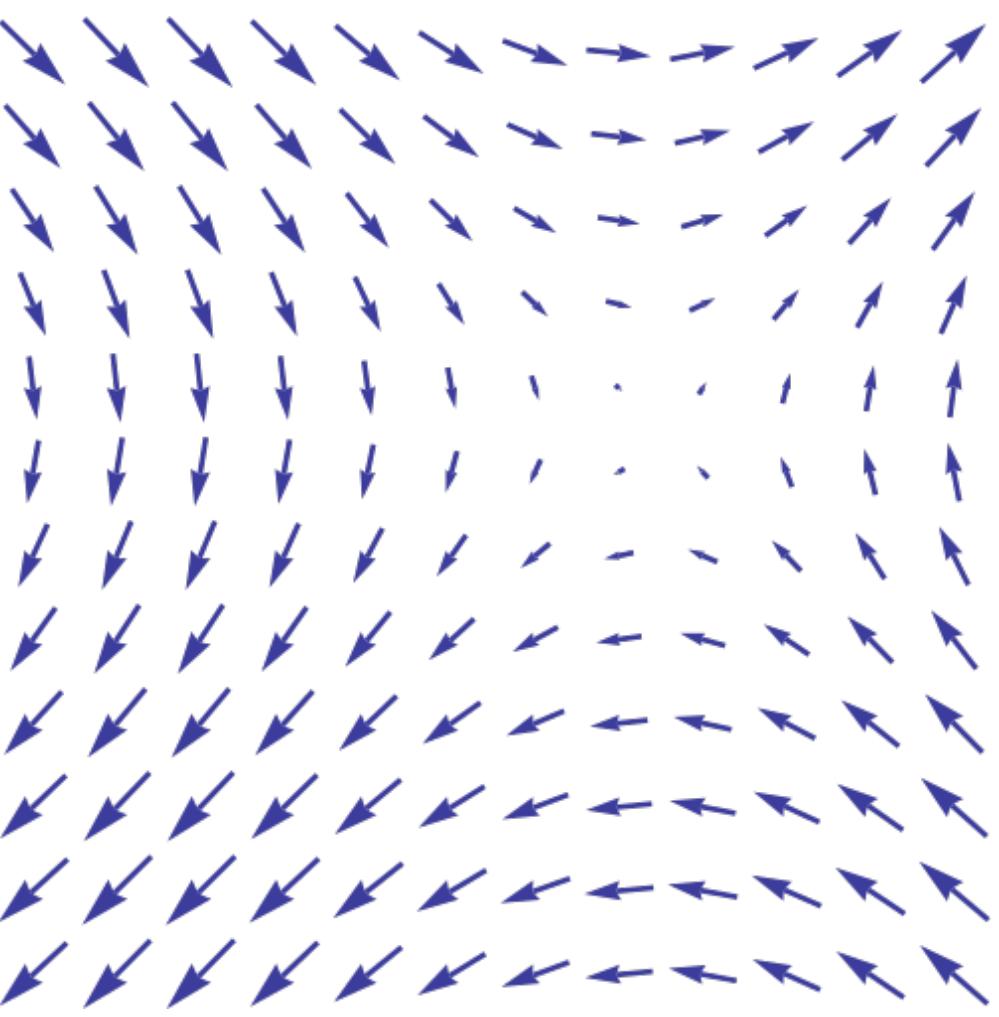
Объемные представления

- Сцена как поверхность - абстракция
 - Удобна в работе и визуализации
 - Иногда не лучший способ описать мир
- Альтернатива - представление сцены через “плотность”
 - Есть ли что-то в данной точке пространства?
 - Сколько света проходить через данную точку?
- Скалярное поле: каждой точке пространства сопоставляем число



Векторные поля

- Встречаются на физике и не только
- Релевантные примеры:
 - Доля поглощенного света
 - Поле расстояний со знаком
 - Фотография



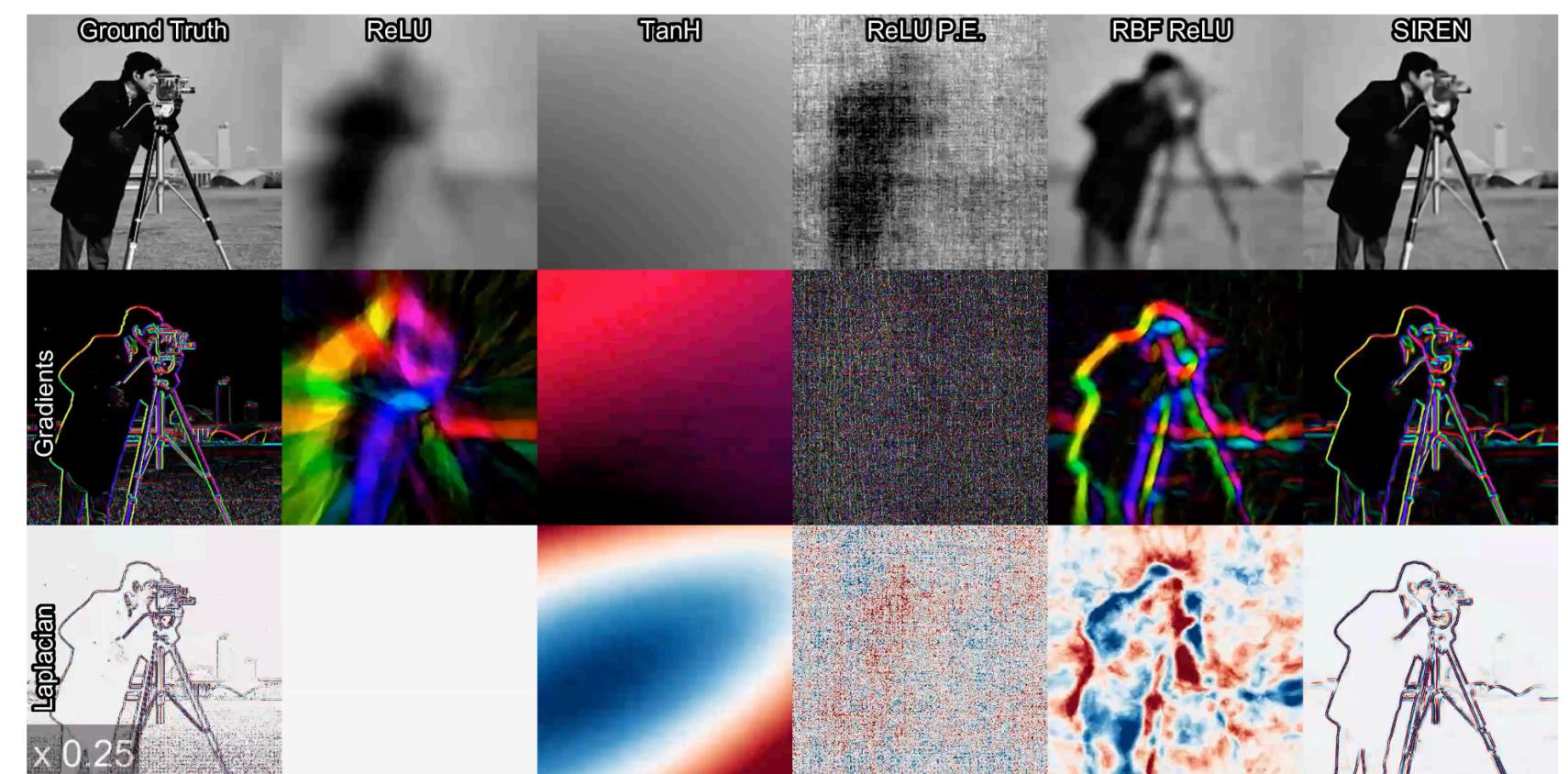
<https://en.wikipedia.org/wiki/X-ray>

<https://arxiv.org/abs/1901.05103>

<https://en.wikipedia.org/wiki/Lenna>

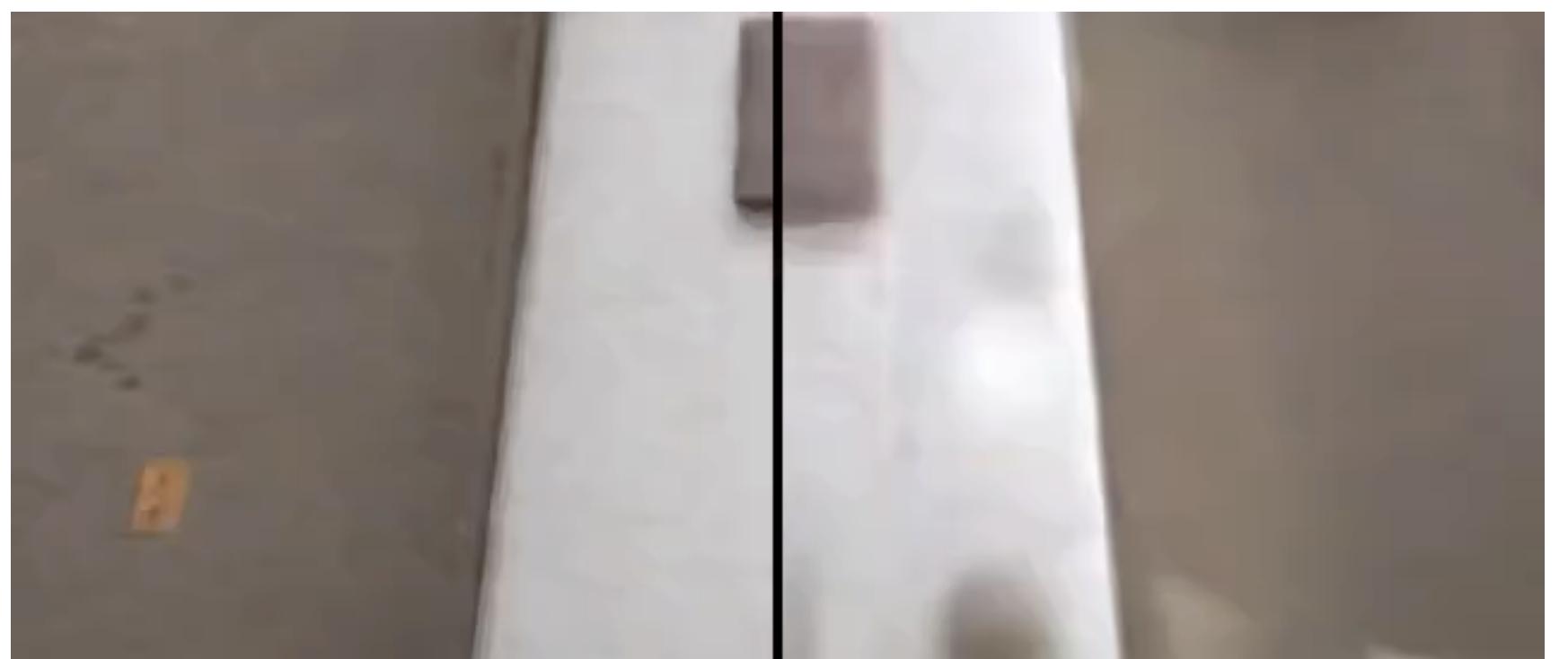
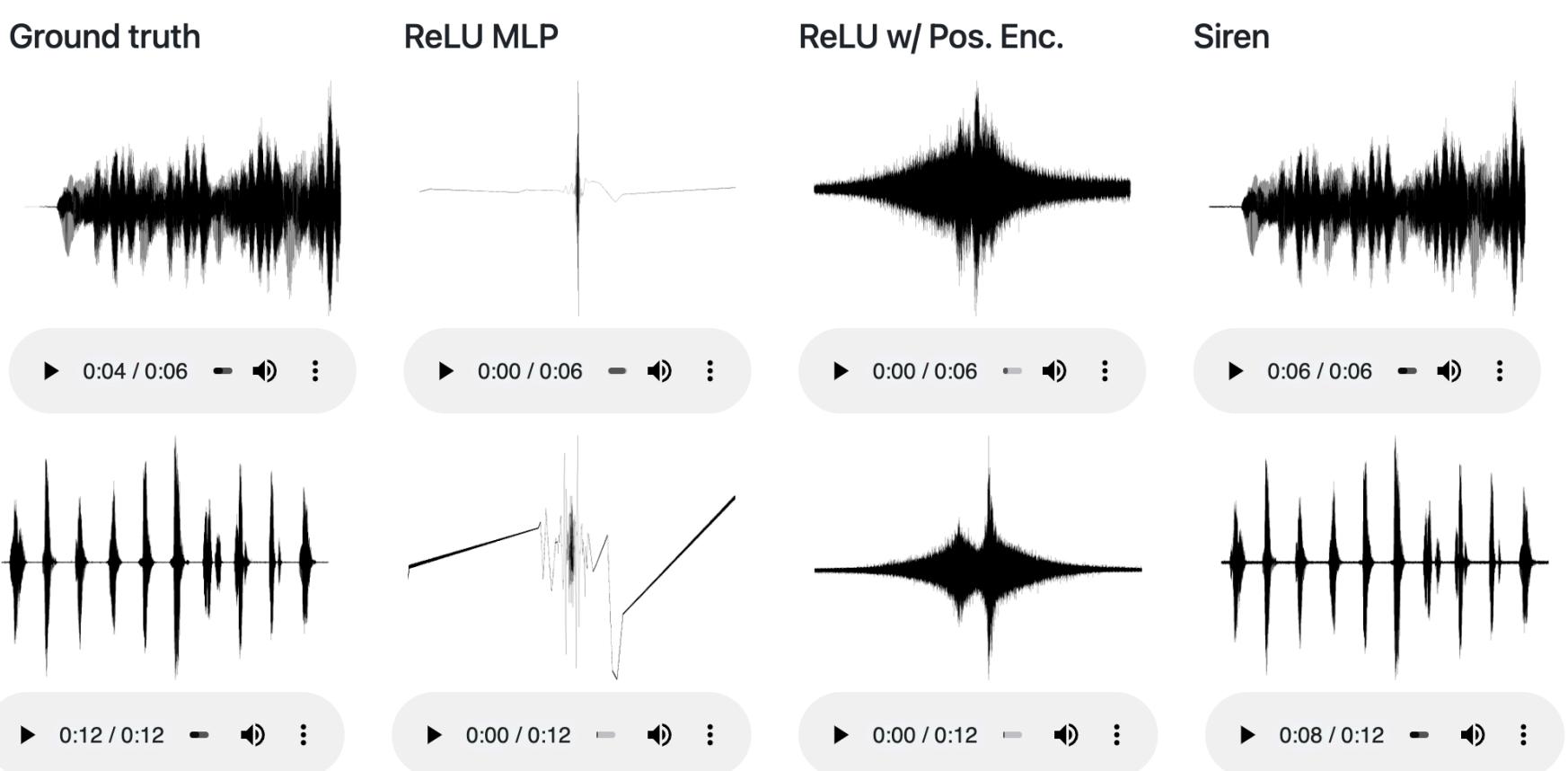
Нейросетевые поля

Вход: точка пространства



Выход: вектор

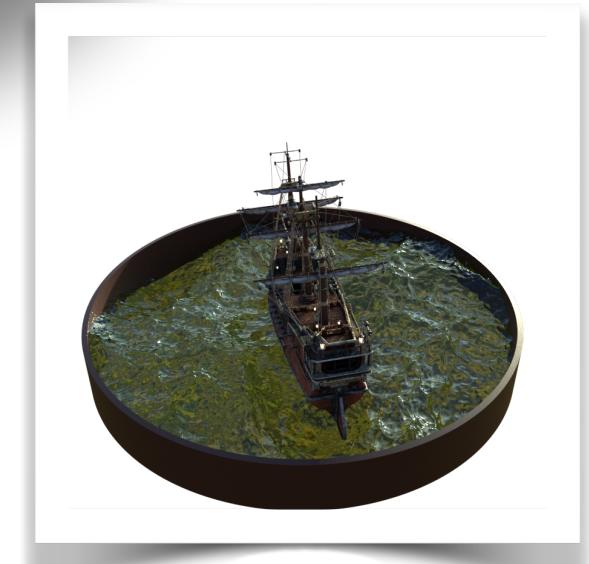
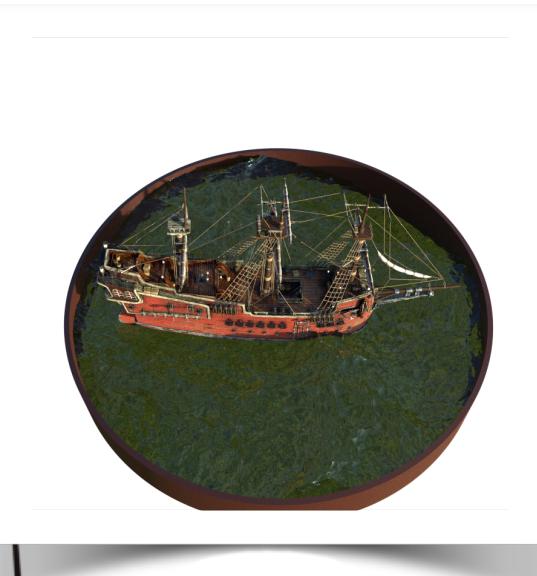
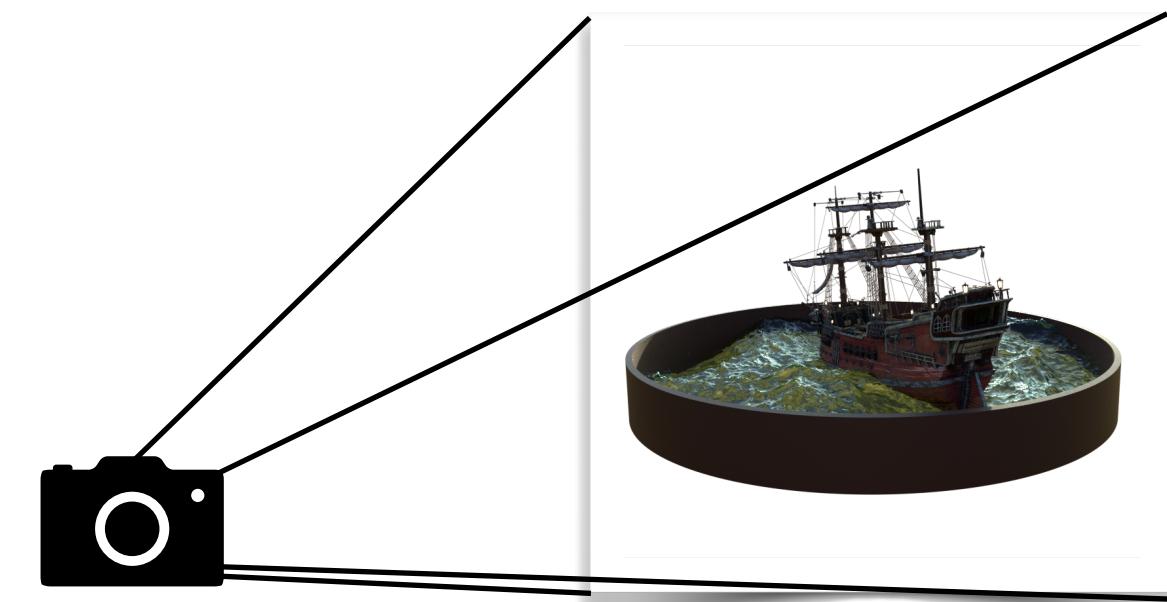
- Архитектура: MLP (+ правильные активации)
- Примеры применения
 - Фото
 - Видео
 - Аудио



Neural Radiance Fields

Нейросетевые поля для реконструкции сцен

- Дано:
 - Набор фотографий сцены с известных ракурсов
 - Тестовый ракурс
- Найти:
 - Изображение сцены, полученное с тестового ракурса



Neural Radiance Fields

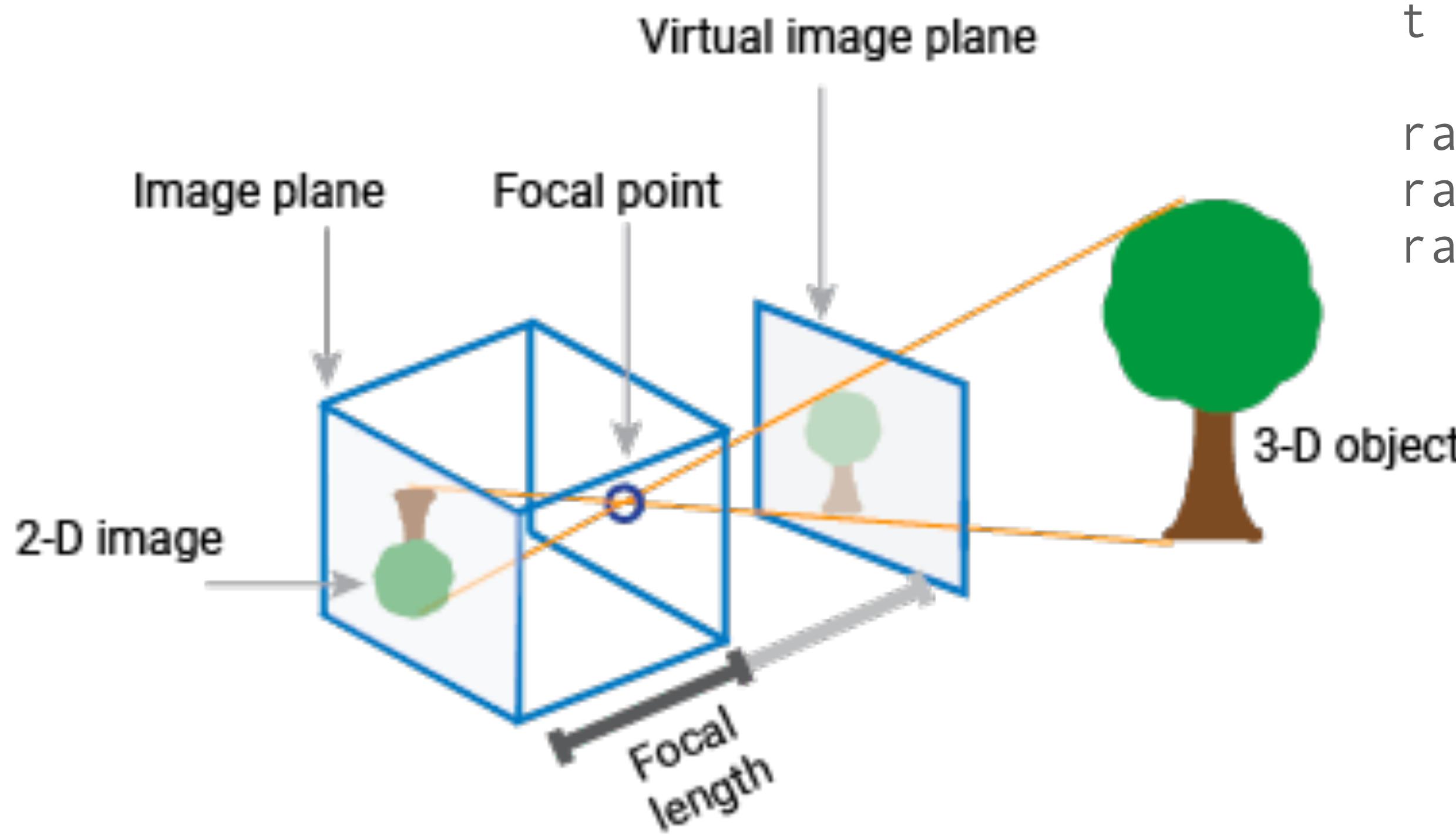
Нейросетевые поля для реконструкции сцен

- Представим сцену с помощью двух полей
 - Плотность: $\sigma(x) : \mathbb{R}^3 \rightarrow \mathbb{R}^+$
 - Светимость: $C(x, d) : \mathbb{R}^3 \times S^2 \rightarrow \mathbb{R}^3$
 - Оно же **Radiance**
- Плотность задает непрозрачность в точке
- Светимость задает цвет в точке
- Архитектура: MLP + позиционное кодирование



Как устроено изображение?

Модель камеры



Генерируем точку на луче:

$(x, y) \in [-1, 1]^2$ - координаты пикселя

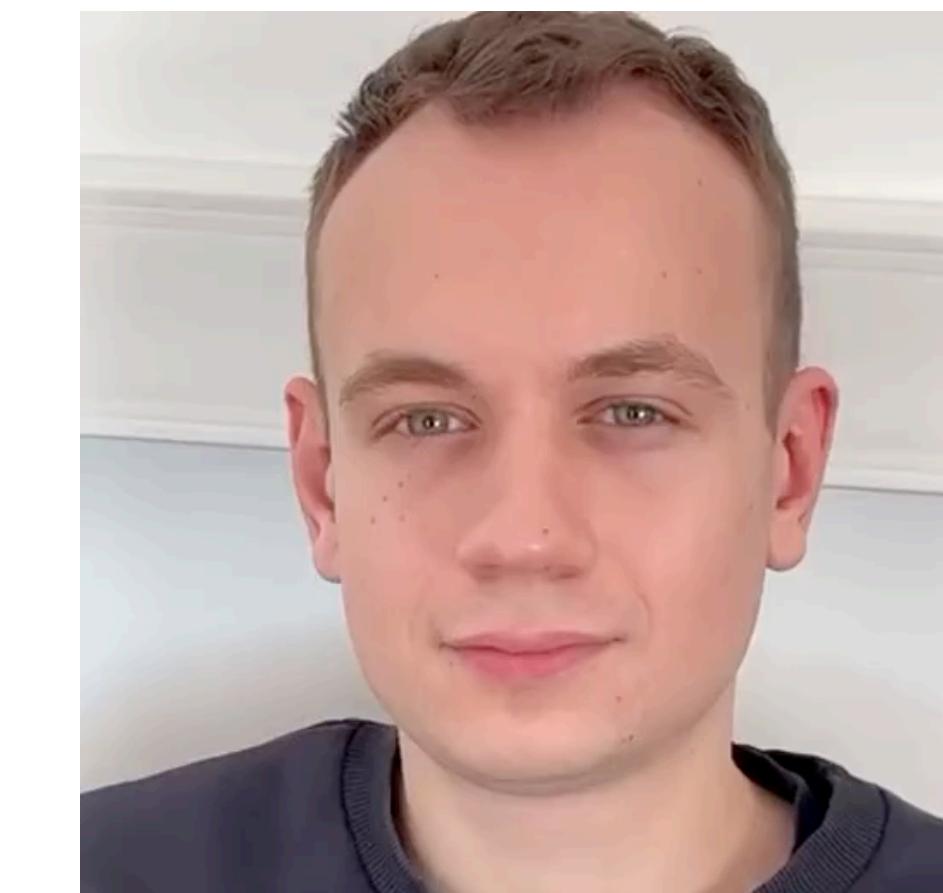
f - фокусное расстояние

t - расстояние до точки на луче от камеры

`ray_origin = (0, 0, 0)`

`ray_direction = normalize((x, y, f) - ray_origin)`

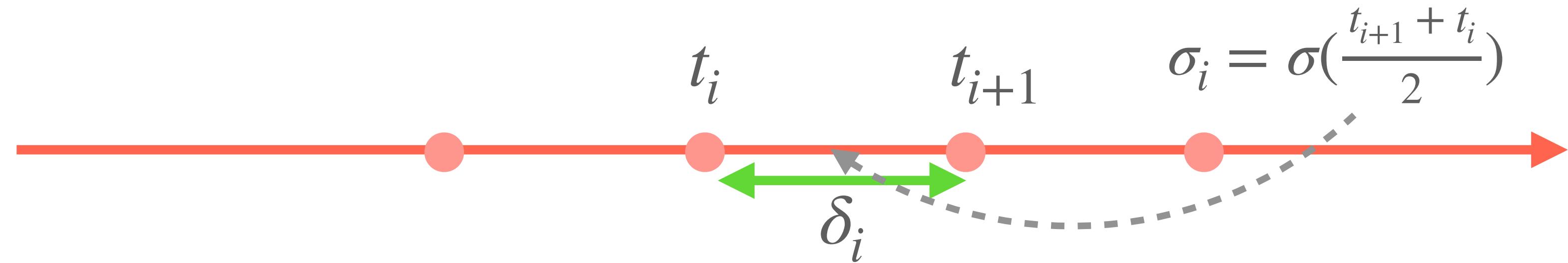
`ray_point = ray_origin + t * ray_direction`



Подсчет цвета пикселя

- Плотность $\sigma(t) \in [0, +\infty)$ задает непрозрачность в t
- Разобьем луч на точки t_1, \dots, t_n и определим

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



- Средний цвет вдоль луча вычисляется по формуле

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

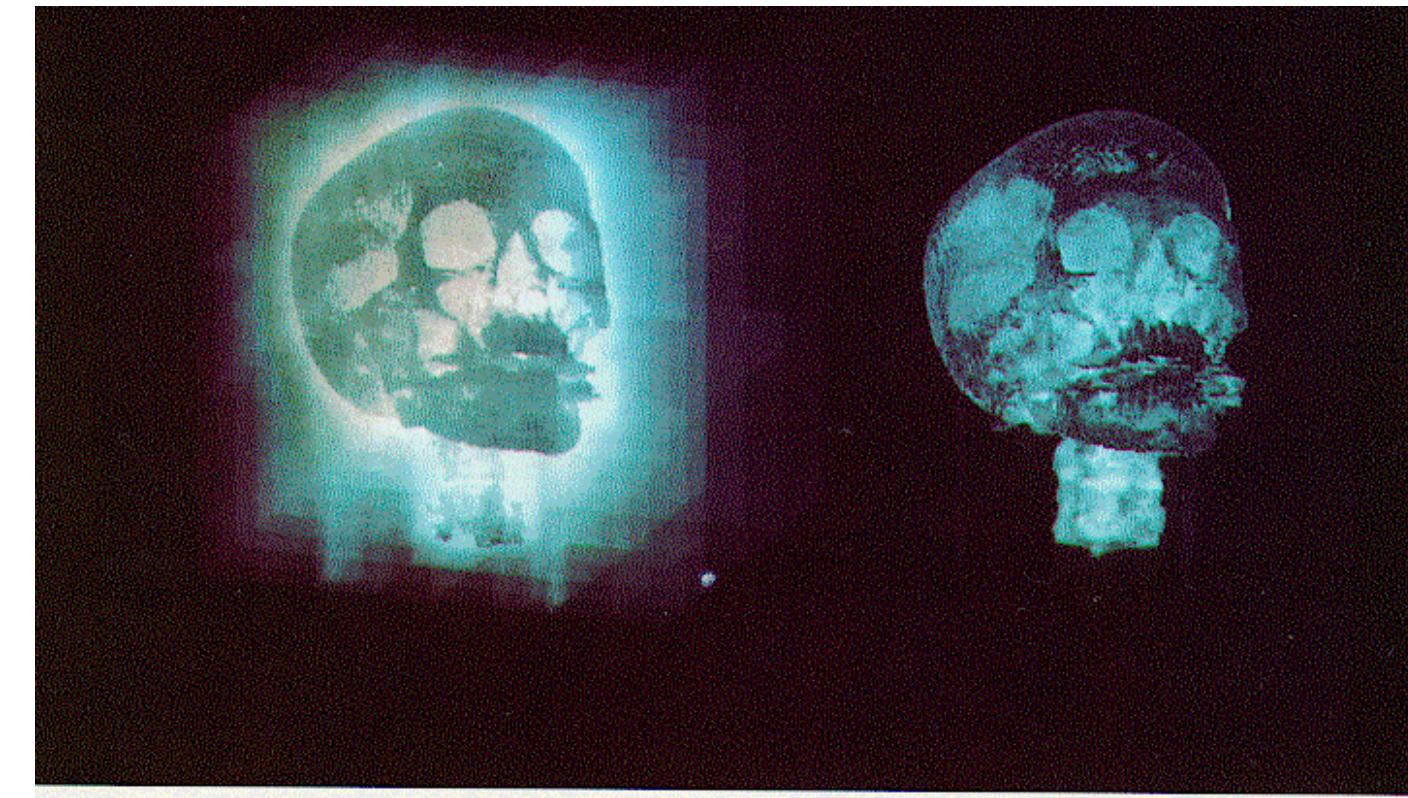


Fig. 11. Costs of rendering Figure 8 using hierarchical enumeration and adaptive termination.

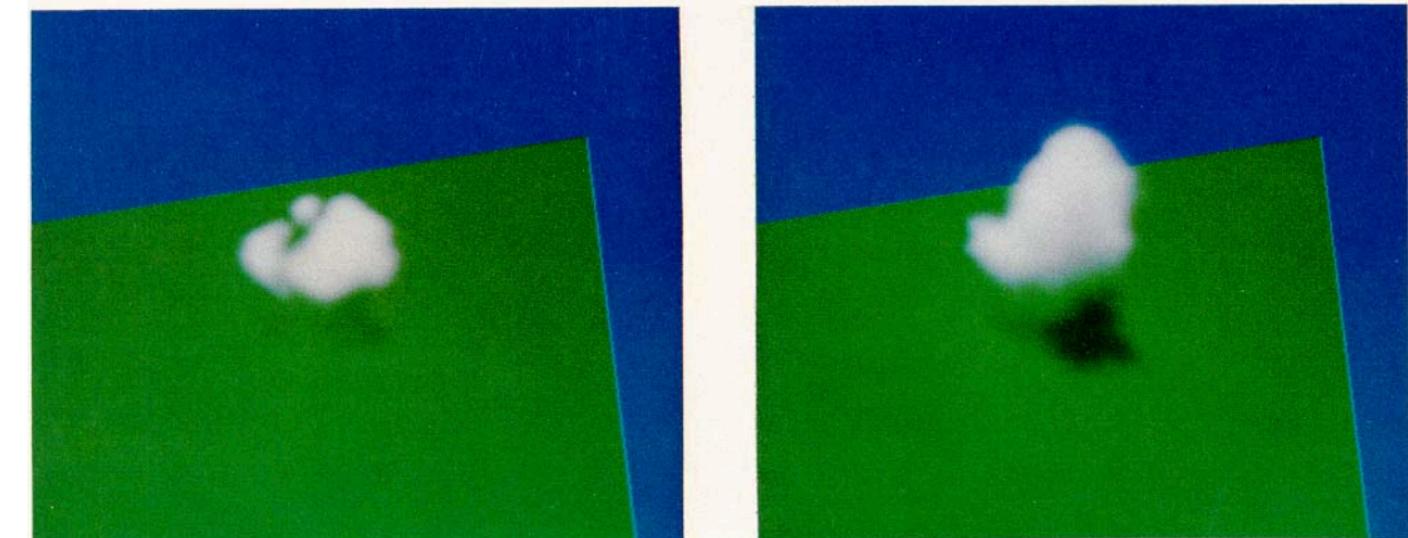


Fig. 5

Fig. 8

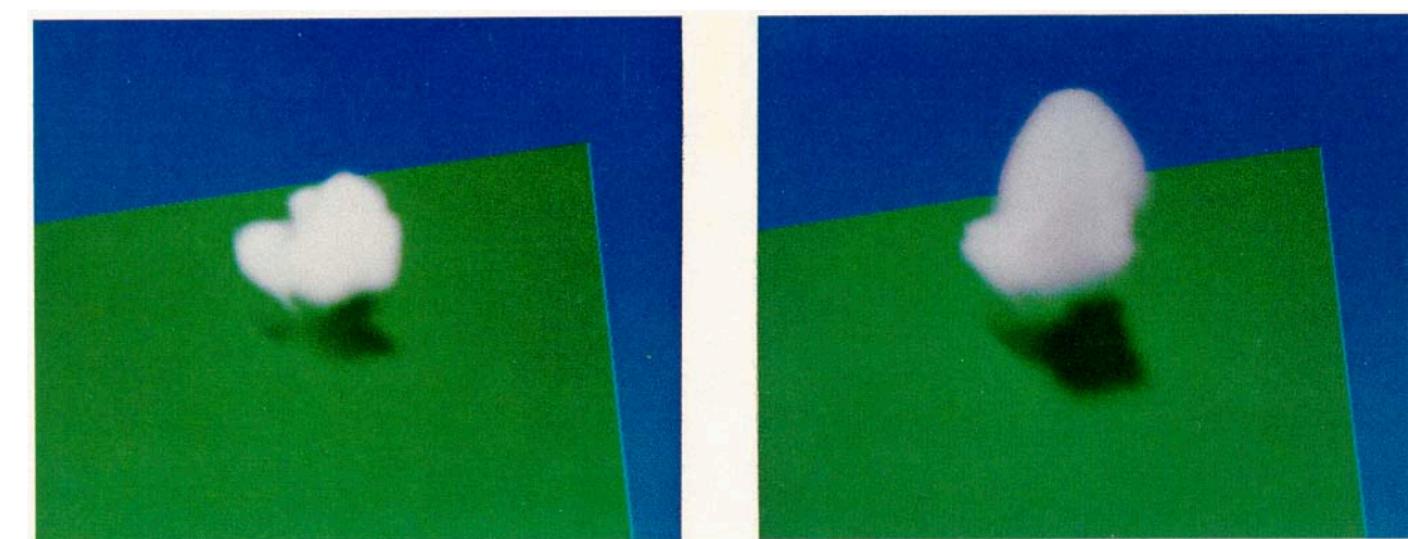


Fig. 6

Fig. 9

Neural Radiance Fields

Обучение

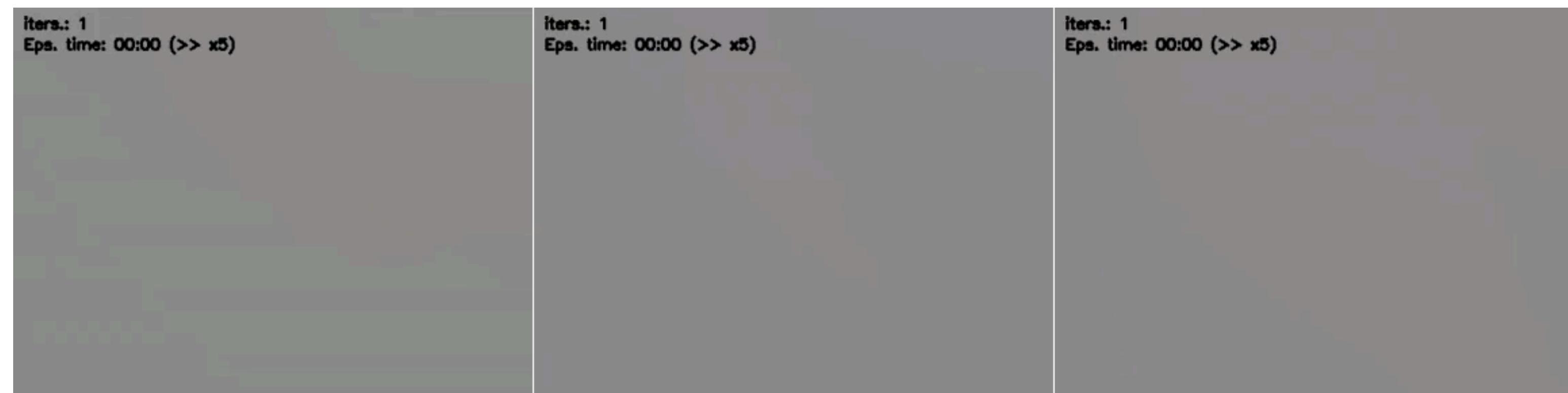
- Обучение: минимизируем среднюю ошибку $\mathbb{E}_D \|C - C_{gt}\|^2$
- $C = \sum_i C(x_i, d) \alpha_i (\prod_{j < i} (1 - \alpha_j))$
- По светимости C и плотности σ

Примеры работы

- Синтетические сцены



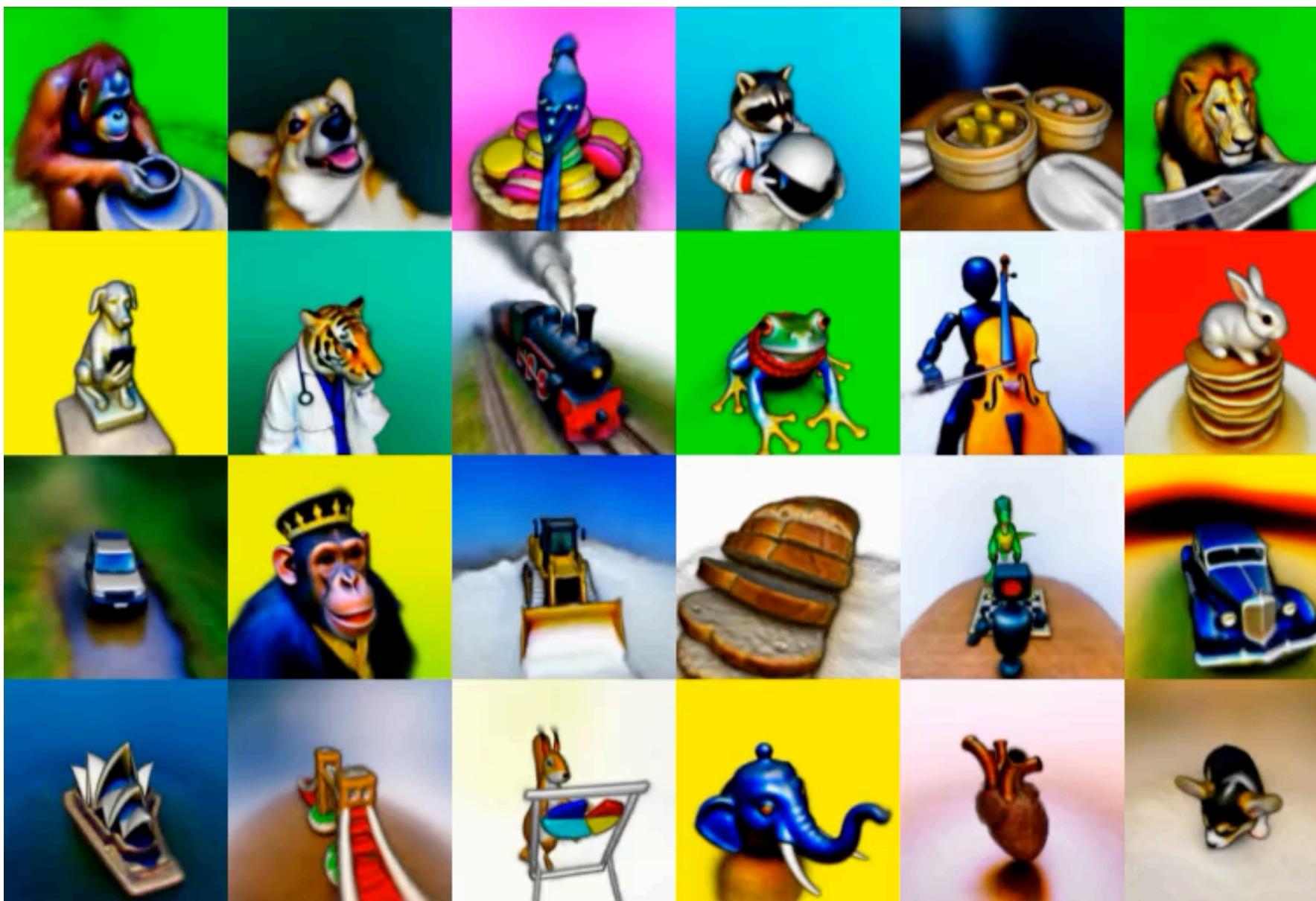
- Фронтальные сцены



Мультимодальные модели

Несколько примеров гибкости решения

- Возможны и более интересные варианты обучающего сигнала
 - Двухмерная диффузионная модель, обусловленная на текст
 - Мультимодальные эмбеддинги для поиска по сцене



Плюсы и минусы подхода

+ Элегантность и гибкость решения

+ Фотореализм

+ Сжатие (~5mb / сцена)

- Не всегда работает на реальных сценах, некорректно поставленная задача

- Долго учится (~48 ч. / сцена)

- Долго рисует (~1 мин. / кадр)



Ускорение модели NeRF

Разреживание

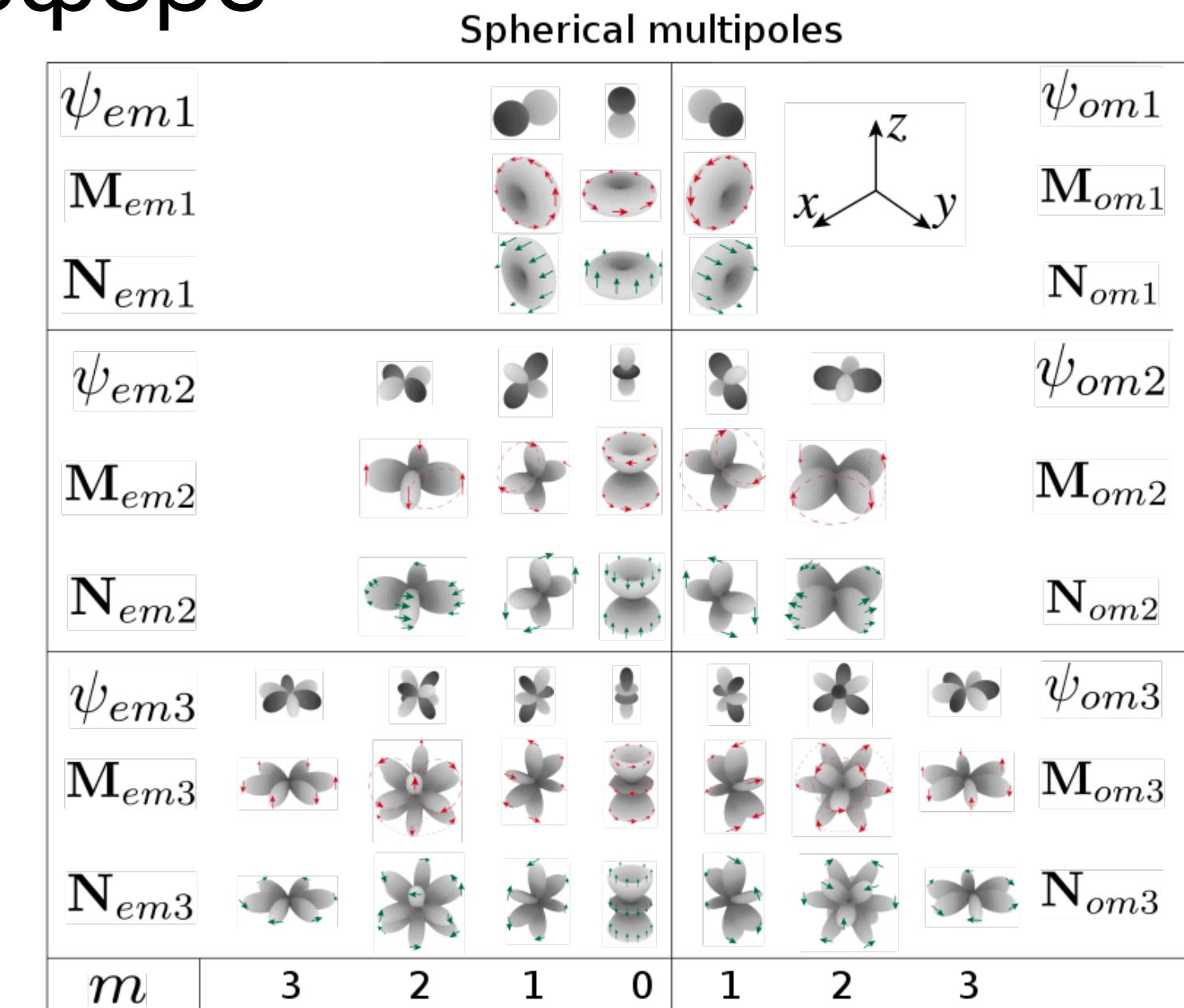
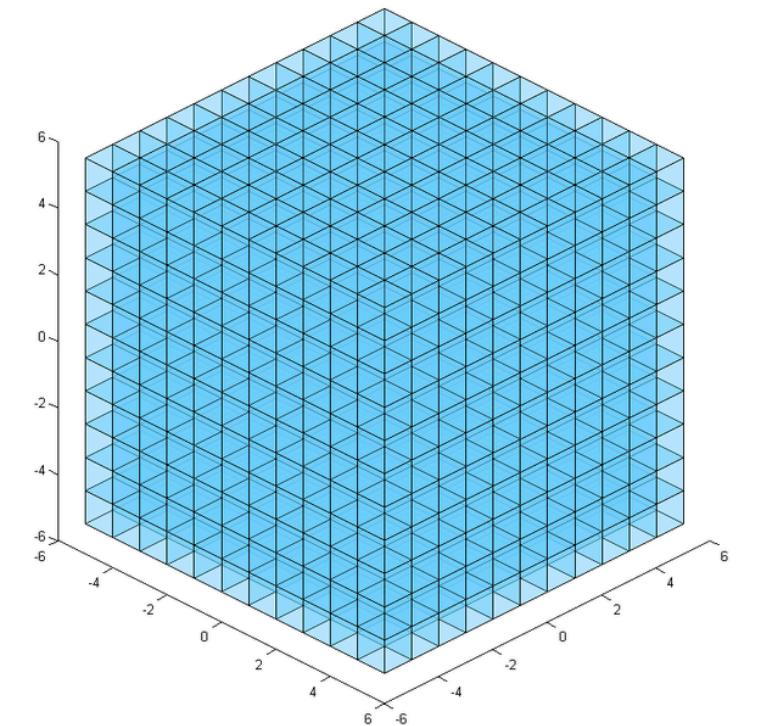
- Каждое слагаемое в сумме для цвета вычисляет значение сети

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

- Если $\sigma(x_i) = 0$, то и $1 - \alpha_i = 0$, слагаемое можно опустить
- Идея: кэширование, не считаем слагаемое i если знаем, что $\sigma(x_i) = 0$
- Результат: ускорение обучения и отрисовки в 10-30 раз

Сеточные представления полей

- Есть и другие подходы к представлению полей
 - Плотность $\sigma(x) : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ можно представить сетки
 - Светимость $C(x, d) : \mathbb{R}^3 \times S^2 \rightarrow \mathbb{R}^3$ подразумевает пятиместную сетку
 - Сферический гармоники - базис среди функций на сфере
- В каждой точке x храним коэффициенты $C(x, \cdot)$
- Время обучения: минуты
- Отрисовка: в режиме реального времени



Гибридный подход

- Есть и другие подходы к представлению полей
 - Плотность $\sigma(x) : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ можно представить сетки
 - Светимость $C(x, d) : \mathbb{R}^3 \times S^2 \rightarrow \mathbb{R}^3$ подразумевает пятимерную сетку
- Гибридное представления $C(x, d) = F(G(x), d)$:
 - $G(x)$ - векторное поле, представленное сеткой
 - $F(v, d)$ - компактная нейронная сеть
- Гибридный подход дает прирост по качеству

Instant-NGP

Плюсы и минусы прошлых подходов

Есть ли компромисс между двумя имеющимися подходами?

MLP архитектура

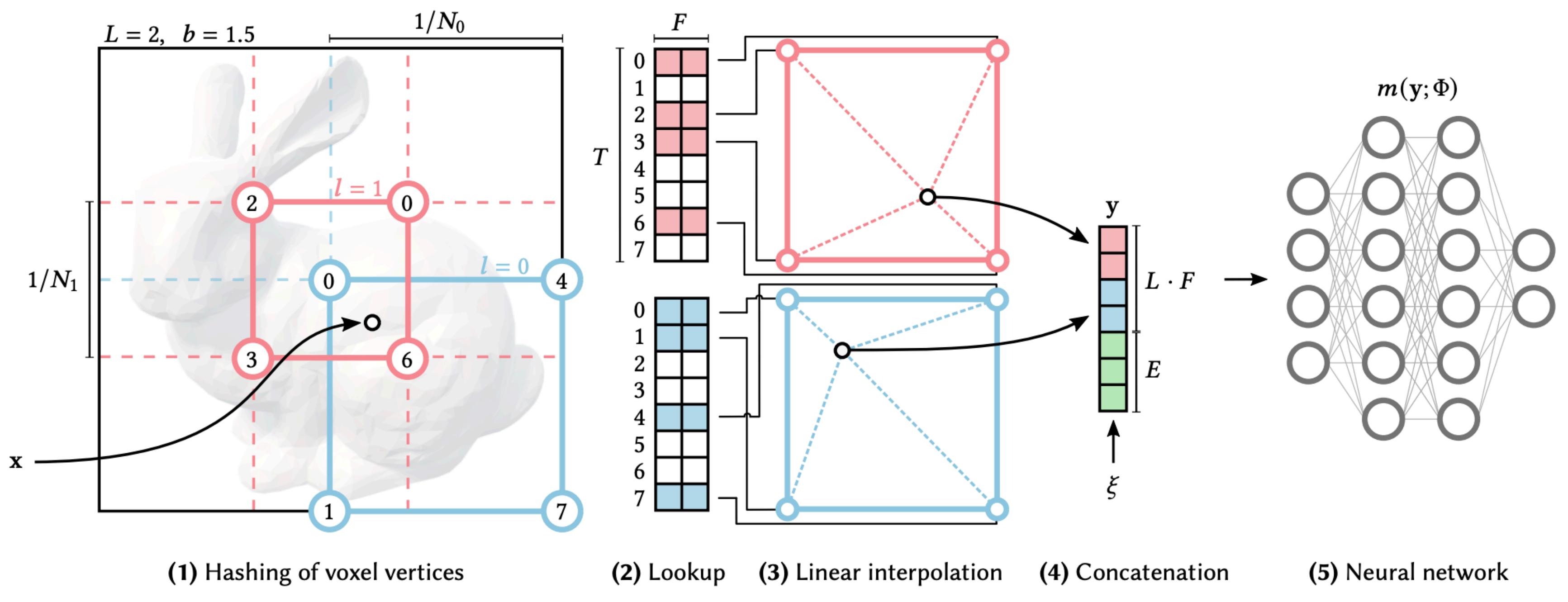
- Размер: ~5mb
- Медленная
- Гибкость связана со скоростью

Трехмерная сетка

- Размер ~1Gb
- Быстрая
- Гибкость связана с памятью

Instant-NGP

HashGrid для скорости и экономии памяти

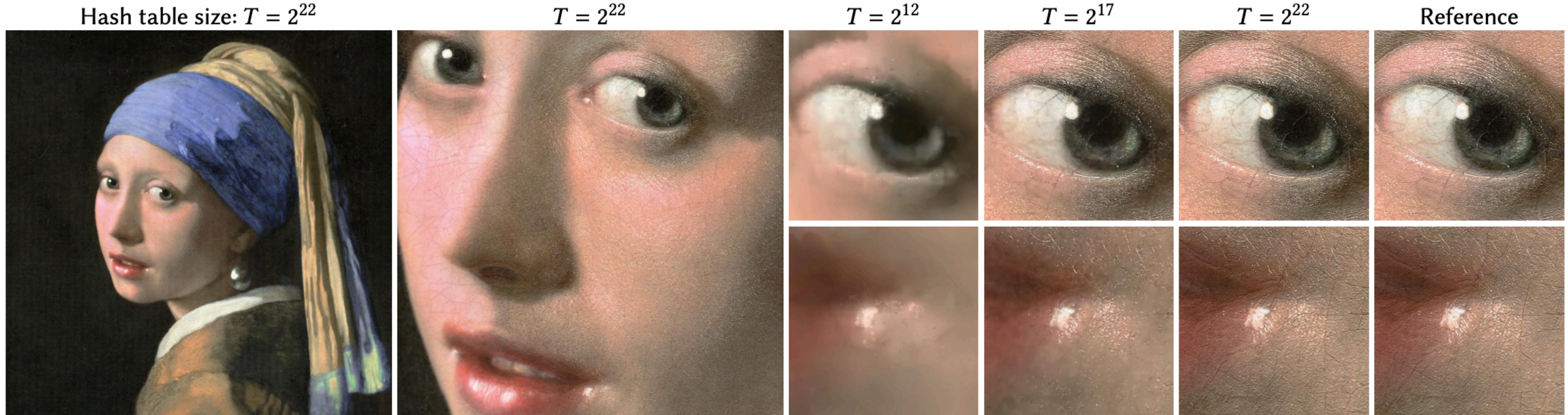


HashGrid

- Размер ~100mb
- Средняя скорость
- Гибкость связана с размером хэш-таблицы
- СО СКОРОСТЬЮ MLP компоненты

Нейросети как графические примитивы

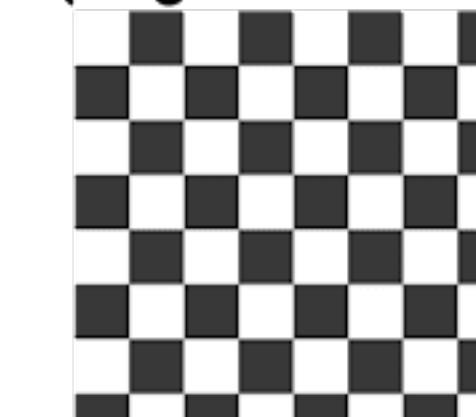
- Вес лучшей модели - 3.4% веса полного изображения (4×10^8 точек)
- PSNR 29.8



Улучшения качества

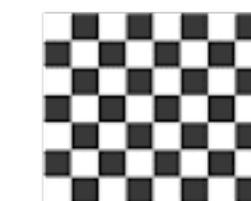
Aliasing

Mip 0
(original texture)



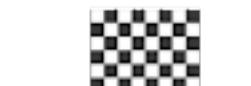
128 x 128

Mip 1



64 x 64

Mip 2



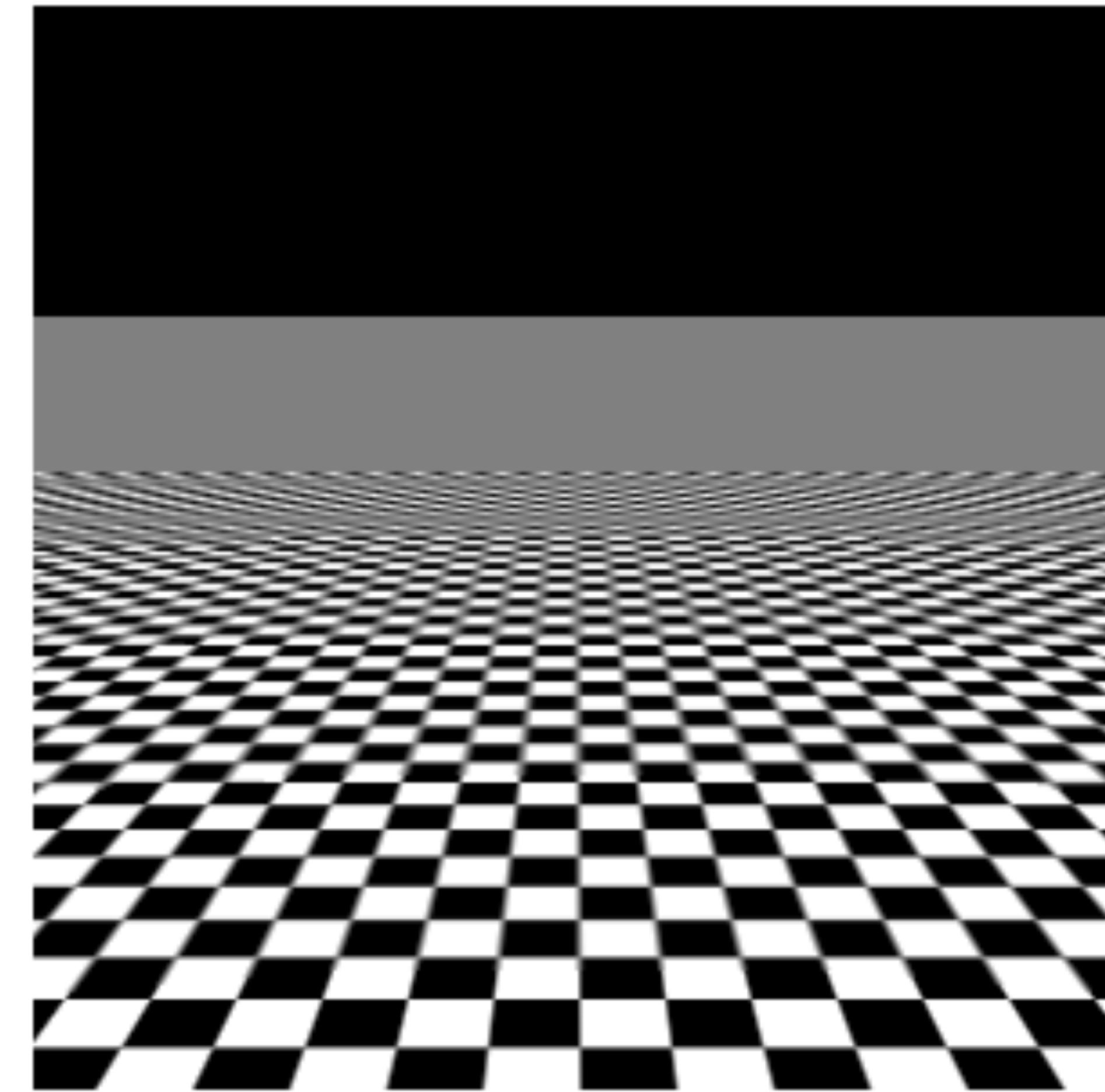
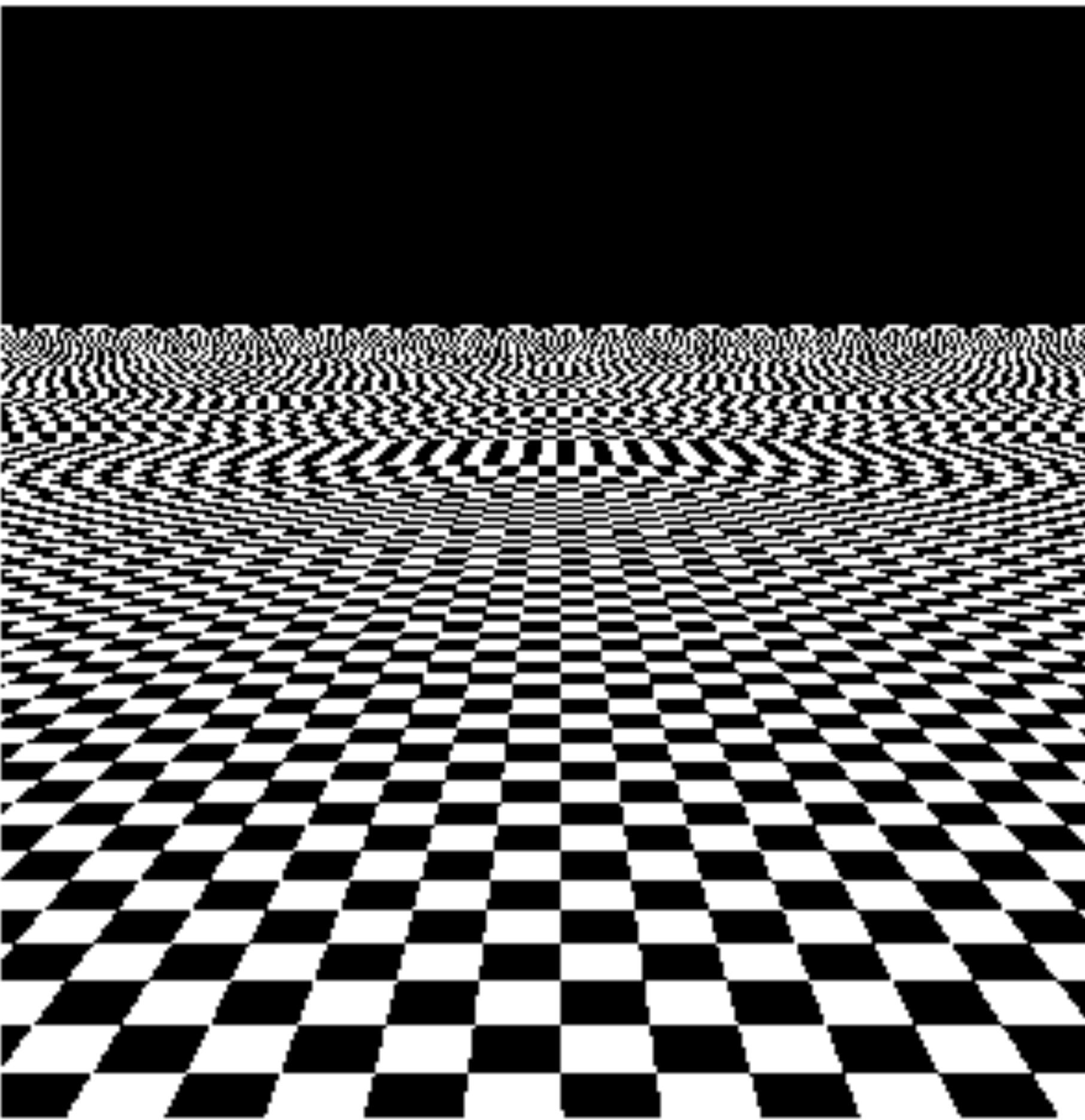
32 x 32

Mip 3

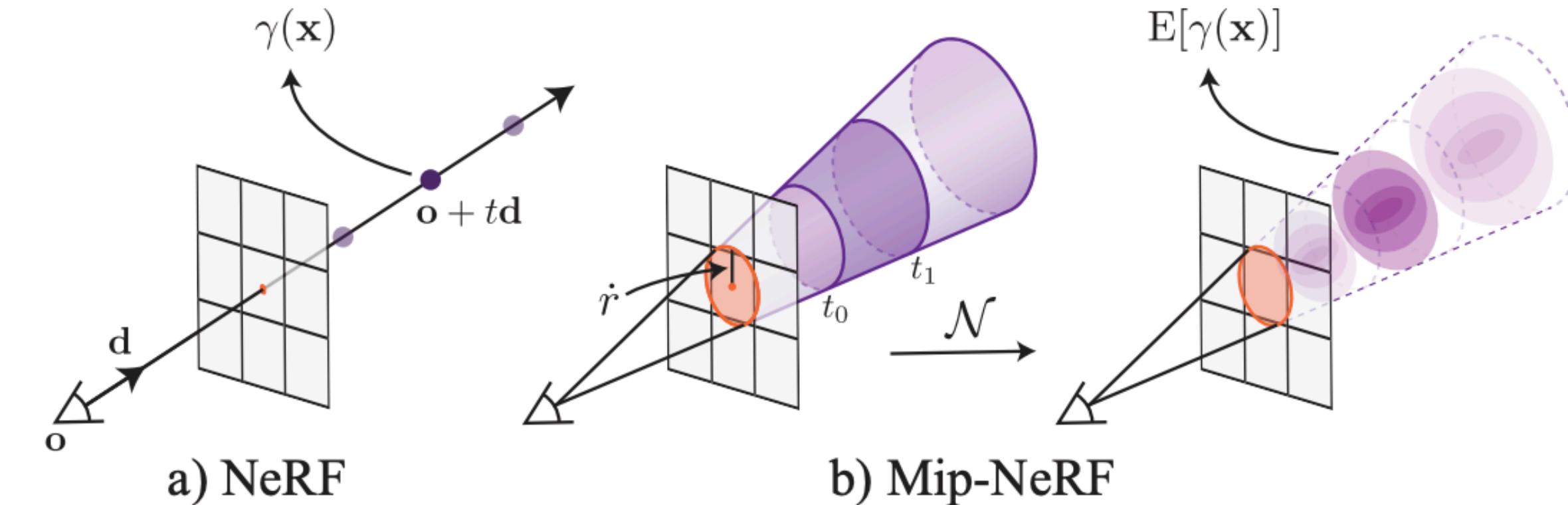


16 x 16

multum in parvo



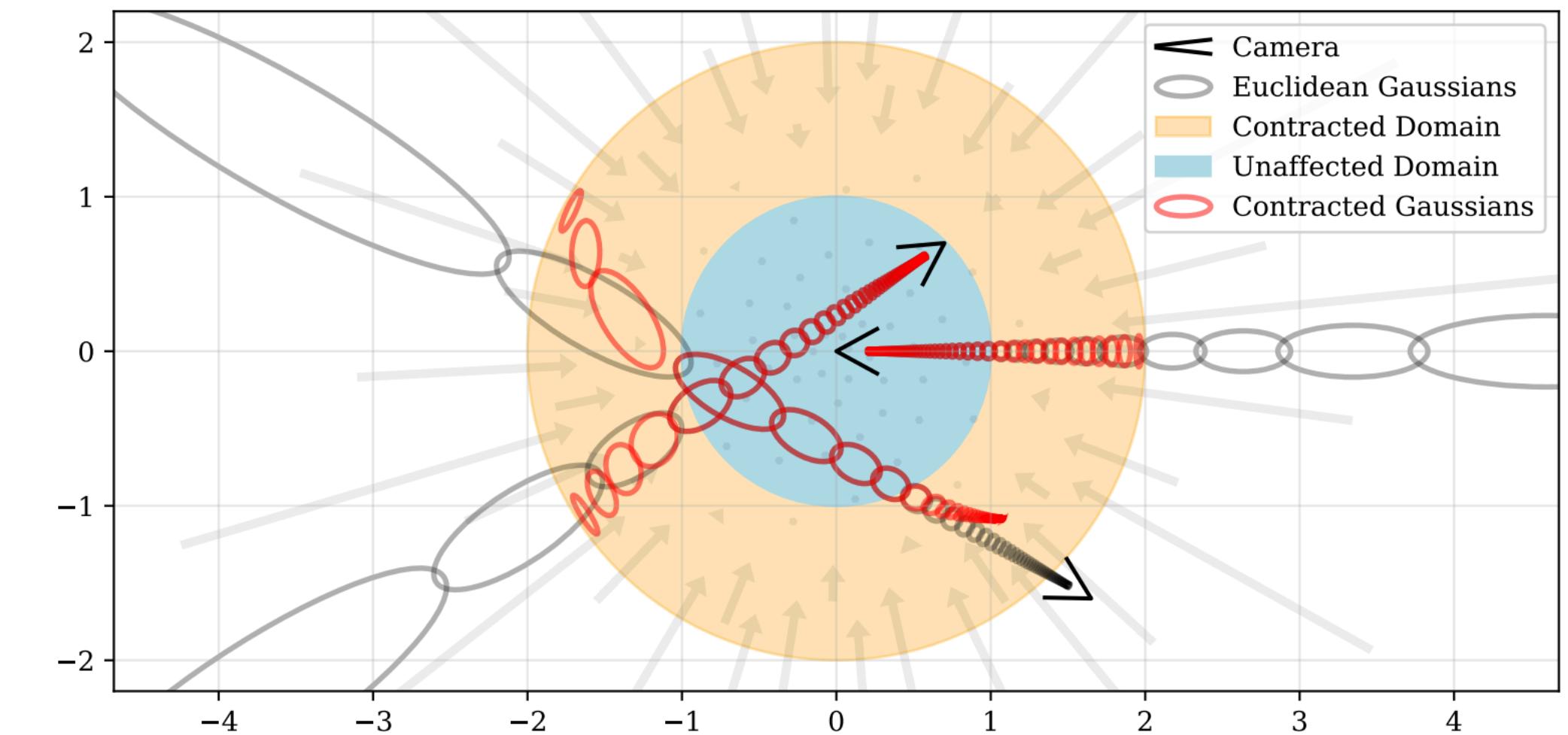
Учет масштаба Параллельная ветвь эволюции



- Оригинальные NeRF обладал смещением в данных
 - Камера на одном расстоянии от сцены
 - На практике у пикселя есть физический размер
 - Идеальный вариант - усреднять выход сети по объему
- Mip-NeRF предложил способ приблизить среднее без дополнительных затрат
- Идея: усреднять представление входа сети вместо выхода

Неограниченные сцены

- Оригинальные NeRF обладал и другим смещением в данных
 - Сцены ограничены по глубине
- NeRF++ отдельно строит представление фона
- Mip-NeRF-360 все пространство в шар
 - Центр сцены остается без изменений
 - Окраины лежат на краю шара



Zip-NeRF

- Две ветки исследования
 - Ускорение: Instant-NGP
 - Качество: Mip-NeRF-360
- Mip-карты работают с MLP
- Zip-NeRF адаптация Mip-карт для Instant-NGP



Что еще можно посмотреть?

- Инструменты:
 - Если хочется обучить NeRF самим: <https://docs.nerf.studio/>
 - Реконструкция сцен за деньги в облаке: AI <https://lumalabs.ai/>
- Не успели обсудить:
 - NeRF для видео: <https://dynibar.github.io/>
 - Реконструкция поверхностей: <https://research.nvidia.com/labs/dir/neuralangelo/>
 - Gaussian Splatting: самая свежая альтернатива NeRF <https://gsplat.tech/>