R^2-Gaussian: Rectifying Radiative Gaussian Splatting for Tomographic Reconstruction

Ruyi Zha[1] Tao Jun Lin[1] Yuanhao Cai[2] Jiwen Cao[1]
Yanhao Zhang[3] Hongdong Li[1]

[1]The Australian National University

[2]Johns Hopkins University [3]University of Technology Sydney

NeurIPS 2024

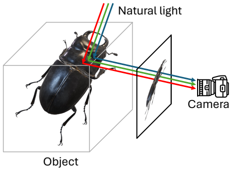
Australian National University

JOHNS HOPKINS UNIVERSITY

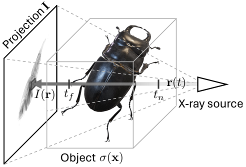
UTS UNIVERSITY OF TECHNOLOGY SYDNEY

NEURAL INFORMATION PROCESSING SYSTEMS

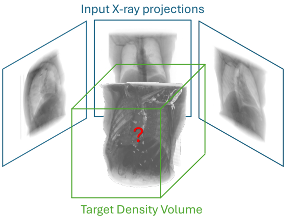
- X-ray: penetrable high-energy beams
- X-ray projection: visualize internal structures
- X-ray imaging function: $I(\mathbf{r}) = \int_{t_n}^{t_f} \sigma(\mathbf{r}(t))dt$
- Tomographic reconstruction
  - Recover the density volume $\sigma(\mathbf{x})$ from multi-angle projections
  - Challenges: sparse-view, noise, speed, etc



RGB imaging

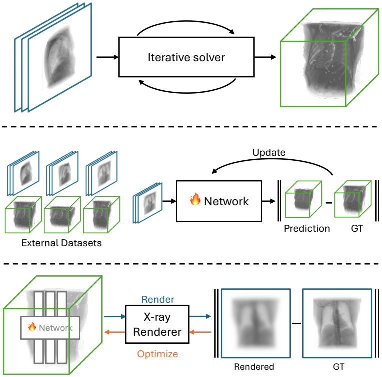


X-ray imaging



Previous Work

- Traditional methods
  - Fast (< 10 min)
  - Bad quality
- Supervised-learning methods
  - Good quality
  - External dataset required
  - Poor generation
- NeRF-based methods
  - Good quality
  - Good generation (self-supervised)
  - Slow training (> 30 min)

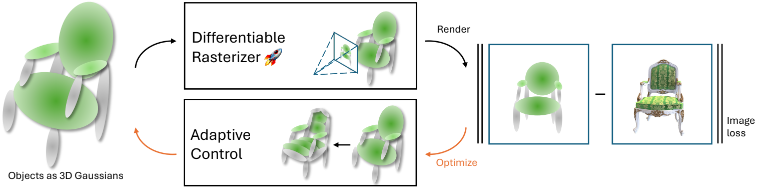


A new method should be...

- Self-supervised: generalizable to arbitrary objects
- Fast training: comparable to traditional methods

Background: 3D Gaussian Splatting (3DGS)

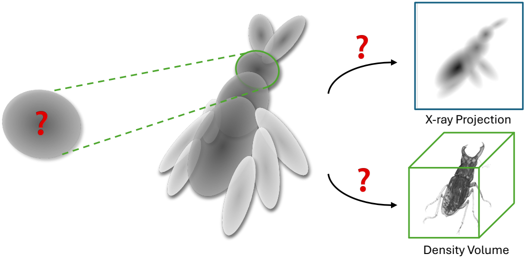
- SOTA method for RGB view synthesis
- Represent objects with trainable 3D Gaussians
- Differentiable rasterization for fast rendering and training



Can we use the idea of 3DGS for tomographic reconstruction?

Challenges

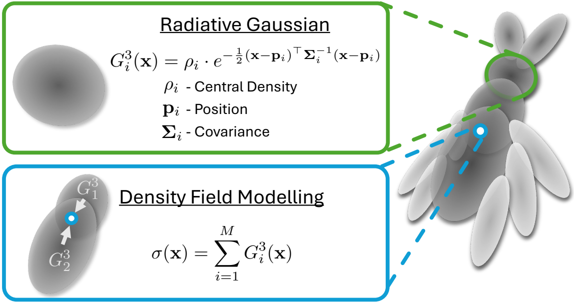
- How to define 3D Gaussians for X-ray imaging?
- How to rasterize X-ray projections?
- How to directly[1] get a density volume from Gaussians?



[1] There are some 3DGS works for X-ray view synthesis, but they can't obtain density volumes from Gaussians, thus not suitable for tomographic reconstruction.

Representing Density Fields as Radiative Gaussians

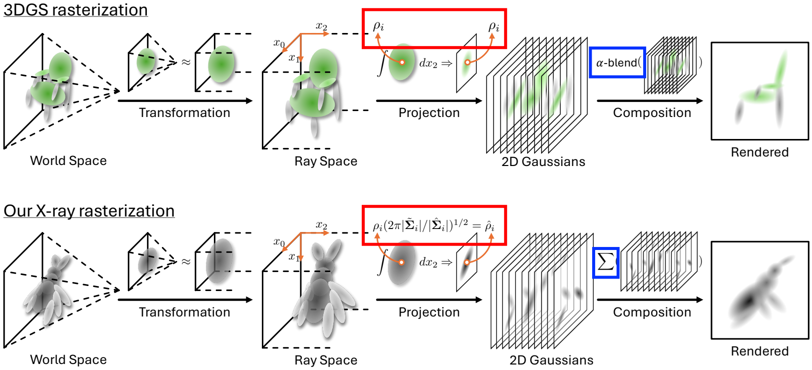
- We define radiative Gaussian as a local Gaussian-shaped density field.
- The overall density field is the sum of radiative Gaussians.
- We use radiative Gasusians for both 2D rendering and 3D querying.



X-ray Rasterization

Rasterization pipeline

- Transformation: world space to ray space
- Projection: 3D Gaussians to 2D Gaussians
- Composition: 2D Gaussians to image



Integration Bias in 3DGS

Integration property of normalized Gaussian distribution

Integrating a normalized 3D Gaussian distribution along an axis yields a normalized 2D Gaussian distribution:

$$\int_{\mathbb{R}} \frac{1}{(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} \exp(-\frac{1}{2}(\mathbf{x}-\mathbf{p})^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{p}))dx_2 = \frac{1}{2\pi|\hat{\mathbf{\Sigma}}|^{\frac{1}{2}}} \exp(-\frac{1}{2}(\hat{\mathbf{x}}-\hat{\mathbf{p}})^\top \hat{\mathbf{\Sigma}}^{-1}(\hat{\mathbf{x}}-\hat{\mathbf{p}})),$$

where $\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} \Rightarrow \hat{\mathbf{x}} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$, $\mathbf{\Sigma} = \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix} \Rightarrow \hat{\mathbf{\Sigma}} = \begin{bmatrix} a & b \\ b & d \end{bmatrix}$, and $\mathbf{p} = \begin{bmatrix} p_0 \\ p_1 \\ p_2 \end{bmatrix} \Rightarrow \hat{\mathbf{p}} = \begin{bmatrix} p_0 \\ p_1 \end{bmatrix}$.

When projecting a 3D Gaussian primitive to its 2D companion, we have:

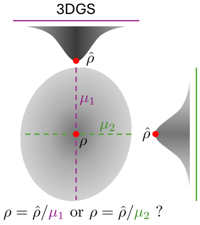
$$\begin{aligned} G^2(\hat{\mathbf{x}}|\hat{\rho}, \hat{\mathbf{p}}, \hat{\mathbf{\Sigma}}) &= \int G^3(\mathbf{x}|\rho, \mathbf{p}, \mathbf{\Sigma})dx_2 = \rho \int \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{p})^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{p})\right)dx_2 \\ &= \rho(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}} \int \frac{1}{(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{p})^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{p})\right)dx_2 \\ &= \rho(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}} \frac{1}{2\pi|\hat{\mathbf{\Sigma}}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\hat{\mathbf{x}}-\hat{\mathbf{p}})^\top \hat{\mathbf{\Sigma}}^{-1}(\hat{\mathbf{x}}-\hat{\mathbf{p}})\right) \\ &= G^2(\hat{\mathbf{x}}|\underbrace{\sqrt{2\pi|\mathbf{\Sigma}|/|\hat{\mathbf{\Sigma}}|}}_{\mu}\rho, \hat{\mathbf{p}}, \hat{\mathbf{\Sigma}}) \end{aligned}$$

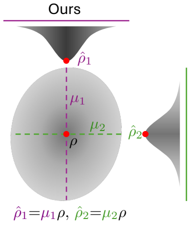
Note the density (opacity) change: $\hat{\rho} = \mu\rho$

Integration Bias in 3DGS

3D density $\Rightarrow$ 2D density: $\hat{\rho} = \underbrace{\sqrt{2\pi|\mathbf{\Sigma}|/|\hat{\mathbf{\Sigma}}|}}_{\mu}\,\rho$

- μ is view-dependent, making $\hat{\rho}$ view-dependent.
- 3DGS: view-independent $\hat{\rho} = \rho$
  - OK for 2D rendering
  - inconsistency (bias) for 3D querying: $\rho \overset{?}{=} \hat{\rho}/\mu$
- Ours: view-dependent $\hat{\rho} = \mu\rho$
  - No inconsistency problem for 3D querying

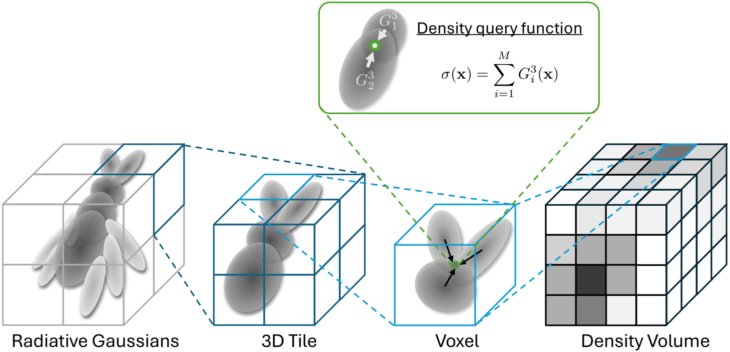




X-ray Voxelization

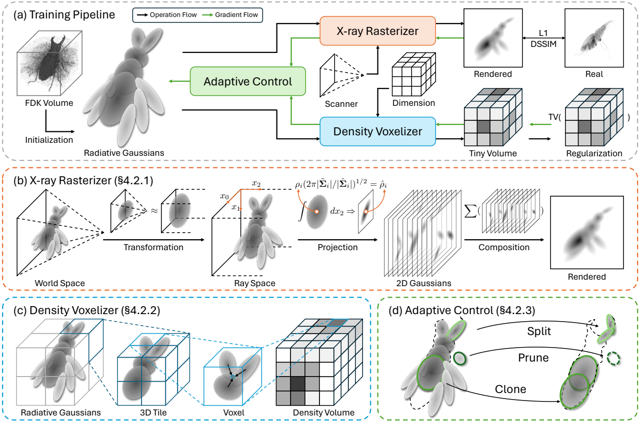
We develop a voxelizer to query density volumes from Gaussians.

- CUDA-based, very fast
- Differentiable, support 3D losses



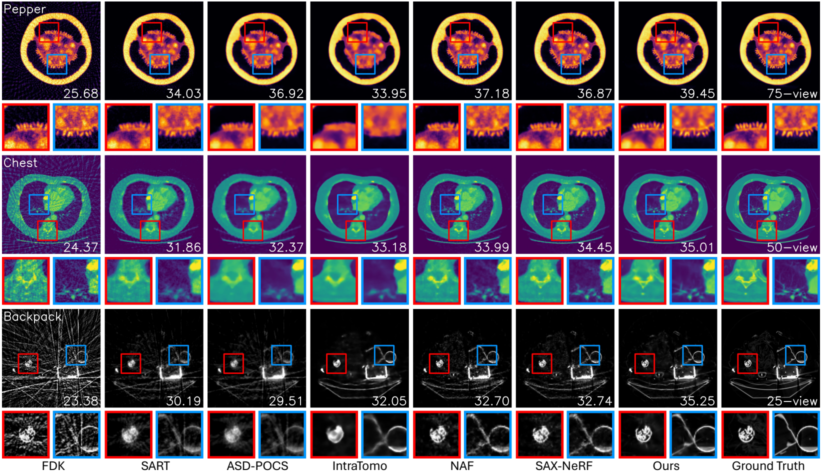
Overall Pipeline

- New initialization strategy by sampling points from FDK volume
- 2D image loss (rasterizer) + 3D regularization (voxelizer)
- Adaptive control for point densification



Experiments: Reconstruction Quality

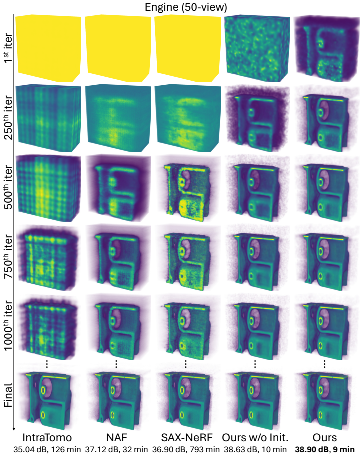
SOTA reconstruction quality



Experiments: Efficiency

- Outperform baselines in 4 minutes (10k iteration)
    - $\approx$ traditional methods
    - $14\times$ faster than NeRF-based methods
- Optimal results in 15 minutes (30k iteration)
    - $4\times$ faster than NeRF-based methods

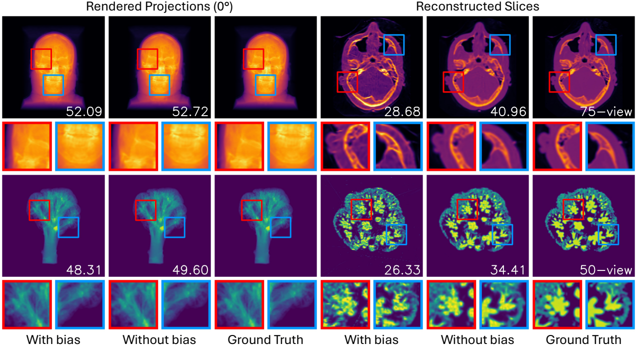
Methods	75-view			50-view			25-view		
	PSNR↑	SSIM↑	Time↓	PSNR↑	SSIM↑	Time↓	PSNR↑	SSIM↑	Time↓
Synthetic dataset									
FDK [13]	28.63	0.497	-	26.50	0.422	-	22.99	0.317	-
SART [2]	36.06	0.897	4m41s	34.37	0.875	3m36s	31.14	0.825	1m47s
ASD-POCS [55]	36.64	0.940	2m25s	34.34	0.914	1m52s	30.48	0.847	56s
IntraTomo [66]	35.42	0.924	2h7m	35.25	0.923	2h9m	34.68	0.914	2h19m
NAF [67]	37.84	0.945	30m43s	36.65	0.932	32m4s	33.91	0.893	31m1s
SAX-NeRF [6]	38.07	0.950	13h5m	36.86	0.938	13h5m	34.33	0.905	13h3m
Ours (iter=10k)	38.29	0.954	2m38s	37.63	0.949	2m35s	35.08	0.922	2m35s
Ours (iter=30k)	38.88	0.959	8m21s	37.98	0.952	8m14s	35.19	0.923	8m28s
Real-world dataset									
FDK [13]	30.03	0.535	-	27.38	0.449	-	23.30	0.335	-
SART [2]	34.42	0.845	5m11s	33.61	0.827	3m28s	31.52	0.790	1m47s
ASD-POCS [55]	36.33	0.868	2m43s	34.58	0.861	1m49s	31.32	0.810	56s
IntraTomo [66]	36.79	0.858	2h25m	36.99	0.854	2h19m	35.85	0.835	2h18m
NAF [67]	38.58	0.848	51m28s	36.44	0.818	51m31s	32.92	0.772	51m24s
SAX-NeRF [6]	34.93	0.854	13h21m	34.89	0.840	13h23m	33.49	0.793	13h25m
Ours (iter=10k)	38.10	0.872	3m39s	37.52	0.866	3m37s	35.10	0.840	3m23s
Ours (iter=30k)	39.40	0.875	14m16s	38.24	0.864	13m52s	34.83	0.833	12m56s



Ablation: Integration Bias

Correcting integration bias benefits both X-ray rendering and reconstruction.

	75-view		50-view		25-view	
	X-3DGS	Ours	X-3DGS	Ours	X-3DGS	Ours
2D PSNR↑	49.97	50.54	47.26	49.70	39.84	46.28
2D SSIM↑	0.987	0.986	0.984	0.986	0.967	0.982
3D PSNR↑	23.40	38.86	21.24	37.98	14.07	35.17
3D SSIM↑	0.660	0.959	0.562	0.952	0.408	0.923



Summary

- First 3DGS-based method for tomographic reconstruction
- SOTA quality and efficiency
- Discovery of integration bias in 3DGS

R^2-Gaussian: Rectifying Radiative Gaussian Splatting for Tomographic Reconstruction

Ruyi Zha[1] Tao Jun Lin[1] Yuanhao Cai[2,*] Jiwen Cao[1]
Yanhao Zhang[3] Hongdong Li[1]

[1]The Australian National University [2]Johns Hopkins University
[3]Robotics Institute, University of Technology Sydney
{ruyi.zha, taojun.lin, jiwen.cao, hongdong.li}@anu.edu.au
caiyuanhao1998@gmail.com yanhao.zhang@uts.edu.au

Abstract

3D Gaussian splatting (3DGS) has shown promising results in image rendering and surface reconstruction. However, its potential in volumetric reconstruction tasks, such as X-ray computed tomography, remains under-explored. This paper introduces R^2-Gaussian, the first 3DGS-based framework for sparse-view tomographic reconstruction. By carefully deriving X-ray rasterization functions, we discover a previously unknown *integration bias* in the standard 3DGS formulation, which hampers accurate volume retrieval. To address this issue, we propose a novel rectification technique via refactoring the projection from 3D to 2D Gaussians. Our new method presents three key innovations: (1) introducing tailored Gaussian kernels, (2) extending rasterization to X-ray imaging, and (3) developing a CUDA-based differentiable voxelizer. Experiments on synthetic and real-world datasets demonstrate that our method outperforms state-of-the-art approaches in accuracy and efficiency. Crucially, it delivers high-quality results in 4 minutes, which is 12× faster than NeRF-based methods and on par with traditional algorithms. Code and models are available on the project page https://github.com/Ruyi-Zha/r2_gaussian.

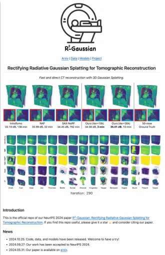
1 Introduction

Computed tomography (CT) is an essential imaging technique for noninvasively examining the internal structure of objects. Most CT systems use X-rays as the imaging source thanks to their ability to penetrate solid substances [20]. During a CT scan, an X-ray machine captures multi-angle 2D projections that measure ray attenuation through the material. As the core of CT, tomographic reconstruction aims to recover the 3D density field of the object from its projections. This task is challenging in two aspects. Firstly, the harmful X-ray radiation limits the acquisition of sufficient and noise-free projections, making reconstruction a complex and ill-posed problem. Secondly, time-sensitive applications like medical diagnosis require algorithms to deliver results promptly.

Existing tomography methods suffer from either suboptimal reconstruction quality or slow processing speed. Traditional CT algorithms [13, 2, 55] deliver results in minutes but induce serious artifacts. Supervised learning-based approaches [32, 33, 10, 35] achieve promising outcomes by learning semantic priors but struggle with out-of-distribution objects. Recently, neural radiance fields (NeRF) [43] have been applied to tomography and perform well in per-case reconstruction [67, 66, 48, 6, 54]. However, they are very time-consuming (> 30 minutes) because a huge amount of points have to be sampled for volume rendering.

*Yuanhao Cai is the corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



R^2-Gaussian

Arxiv | Data | Models | Project

Rectifying Radiative Gaussian Splatting for Tomographic Reconstruction

Fast and direct CT reconstruction with 3D Gaussian Splatting.

Iteration: 290

Introduction

This is the official repo of our NeurIPS 2024 paper R^2-Gaussian: Rectifying Radiative Gaussian Splatting for Tomographic Reconstruction. If you find this repo useful, please give it a star ⭐ and consider citing our paper.

News

- 2024.10.25: Code, data, and models have been released. Welcome to have a try!
- 2024.09.27: Our work has been accepted to NeurIPS 2024.
- 2024.05.31: Our paper is available on arxiv.

Paper

Project page

Australian National University

JOHNS HOPKINS UNIVERSITY

UTS UNIVERSITY OF TECHNOLOGY SYDNEY

NEURAL INFORMATION PROCESSING SYSTEMS