NEURAL INFORMATION PROCESSING SYSTEMS

GVKF: Gaussian Voxel Kernel Functions for Highly Efficient Surface Reconstruction in Open Scenes

Gaochao Song * , Chong Cheng * , Hao Wang

AI Thrust, HKUST(GZ)

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

NEURAL INFORMATION PROCESSING SYSTEMS

Motivation

Method	Characteristic	Pros	Cons
3DGS	Explicit Discrete Alpha Blending	• Fast Rendering	• High Mem Consumption • Hard to Fit Continuous Surface
INR	Implicit Continuous Volume Rendering	• Better Continuous Surface Fitting • Mem Efficient	• Low Rendering Speed due to Dense Sampling

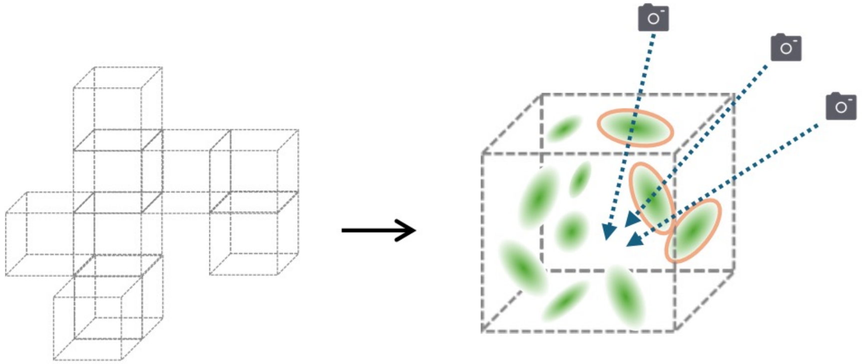
We propose GVKF:

- ✓ Representing Continuous Surface
- ✓ High Rendering Speed, w/o Volume Rendering
- ✓ Implicit, Memory Efficient

NEURAL INFORMATION PROCESSING SYSTEMS

Method

1. Implicit Gaussian Representation



- Gaussian Attributes are stored in sparse voxel grids as 1D latent code $\mathcal{F}$

$$\mathcal{G}(x) = \alpha \cdot e^{-\frac{1}{2}(x-p)^T \Sigma^{-1}(x-p)}$$

- Attributes are decoded by several MLPs conditioned on 1D latent code

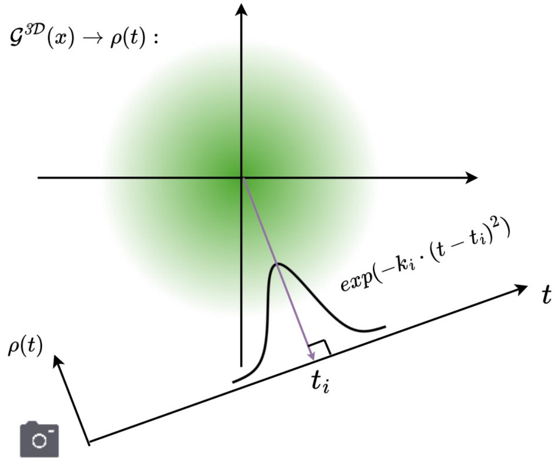
$$\alpha = \mathrm{MLP}_{\alpha}(\mathcal{F}, \mathrm{camera}), R = \mathrm{MLP}_{R}(\mathcal{F}), s = \mathrm{MLP}_{s}(\mathcal{F}), c = \mathrm{MLP}_{c}(\mathcal{F}, \mathrm{camera}).$$

Method

2. Modified 3DGS Rendering

- 3DGS are transformed to 1D Gaussian functions along the camera ray to evaluate the influence of opacity

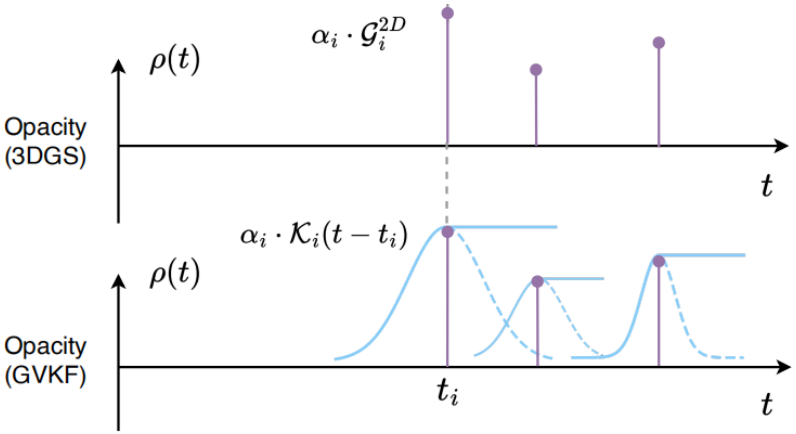
$$t_i = \frac{p^T \Sigma^{-1} v}{v^T \Sigma^{-1} v}$$



- Alpha blending (Equivalent to 3DGS Rendering):

3DGS: $$C = \sum_{i=1}^{N} c_i \cdot \alpha_i \cdot \mathcal{G}_i^{2D} \prod_{j=1}^{i-1} (1 - \alpha_j \cdot \mathcal{G}_j^{2D})$$

GVKF: $$C = \sum_{i=1}^{N} c_i \cdot \alpha_i \cdot \mathcal{K}_i(0) \prod_{j=1}^{i-1} (1 - \alpha_j \cdot \mathcal{K}_j(0))$$

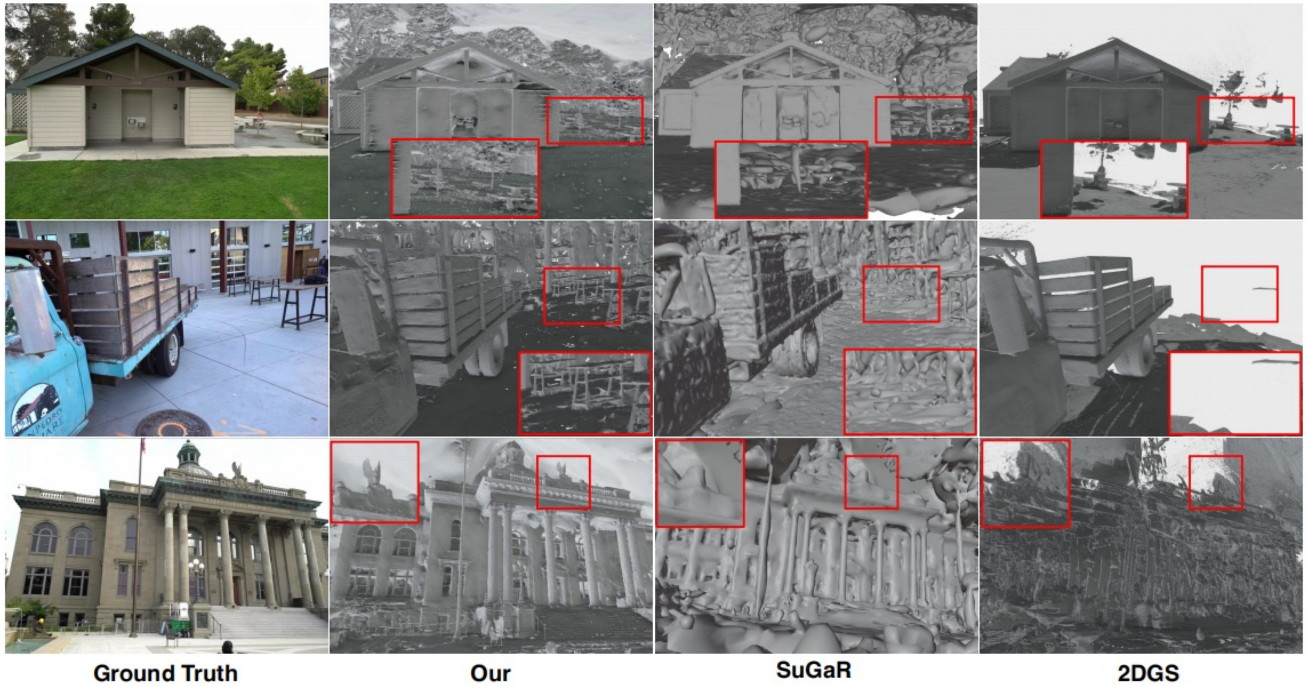


- Surface Representation (CDF of kernel functions):

$$\Phi(t) = \sum_{i=1}^{N} \alpha_i \cdot \mathcal{K}_i(t - t_i) \prod_{j=1}^{i-1} (1 - \alpha_j \cdot \mathcal{K}_j(t - t_j))$$

NEURAL INFORMATION PROCESSING SYSTEMS

Results



Method	PSNR ↑	C-D ↓	MB (Storage) ↓	GB (GPU) ↓	FPS ↑	Training Time ↓
NeuS	13.24	**0.76**	170	31	∼ 0.1	5 h
F^2-NeRF	24.70	886.77	130	24	∼ 0.1	0.8 h
StreetSurf	27.12	1.02	540	22	∼ 0.1	1.5 h
3DGS	27.99	3.57	230	23	**63**	0.75 h
SuGaR	23.71	3.08	228	33	56	1.5 h
2DGS	28.51	1.67	238	23	51	**0.7 h**
GVKF (Ours)	**30.24**	1.57	**30**	**14**	32	1.5 h

Method	Implicit			Explicit			
	NeuS	Geo-NeuS	Neuralangelo	SuGaR	3DGS	2DGS	Ours
Barn	0.29	0.33	**0.70**	0.14	0.13	0.36	**0.40**
Caterpillar	0.29	0.26	**0.36**	0.16	0.08	0.23	**0.34**
Courthouse	0.17	0.12	**0.28**	0.08	0.09	0.13	**0.25**
Ignatius	0.83	0.72	**0.89**	0.33	0.04	0.44	**0.51**
Meetingroom	0.24	0.20	**0.32**	0.15	0.01	0.16	**0.23**
Truck	0.45	0.45	**0.48**	0.26	0.19	0.26	**0.40**
Mean	0.38	0.35	**0.50**	0.19	0.09	0.30	**0.36**
Time	>24 h	>24 h	>24 h	>1 h	∼**15 min**	∼30 min	∼1.5 h

NEURAL INFORMATION PROCESSING SYSTEMS

Thanks !

Project page: https://3dagentworld.github.io/gvkf/