Generalizable 3D-GS: Today and Future

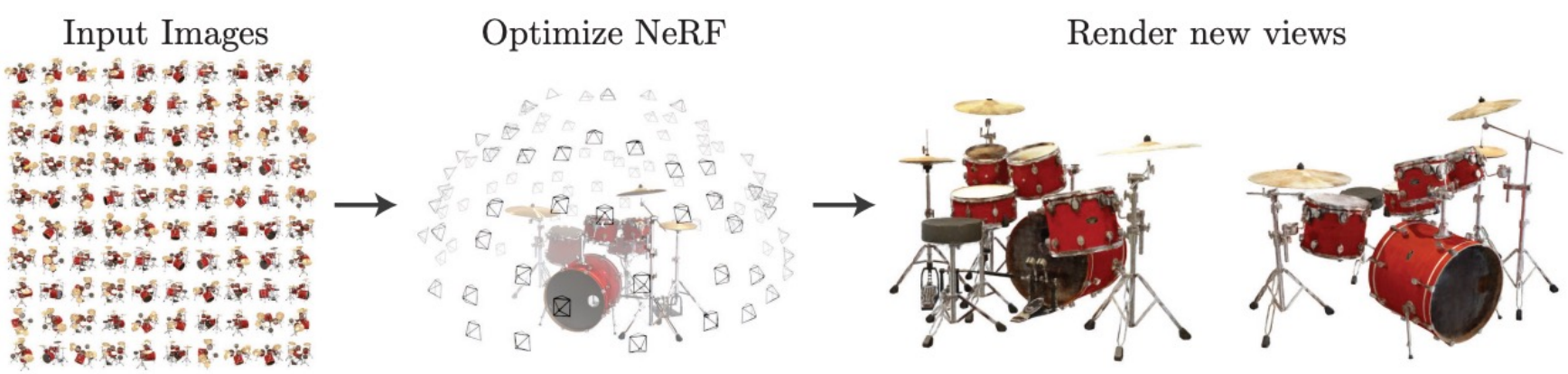
Hao Li
BRAIN Lab, Northwestern Polytechnical University; VIS, Baidu Inc.

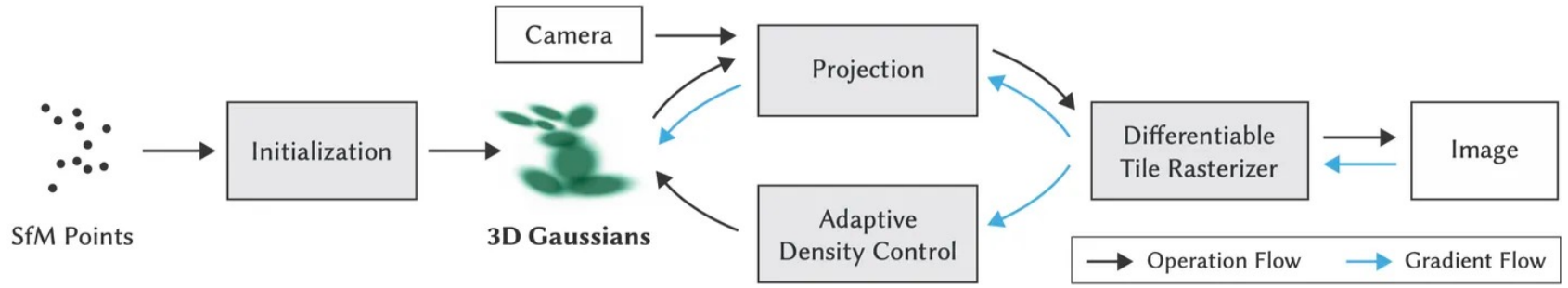
西北工業大學
NORTHWESTERN POLYTECHNICAL UNIVERSITY

2024.07.18

Introduction

Introduction of Vanilla NeRF / 3D-GS





Initialized by COLMAP pointlouds, 3D-GS use learnable Gaussians to represent the 3D scenes / objects, and render the novel view by splatting the Gaussians into the novel psoes.

Rendered Video.

Vanilla NeRF / Gaussian-Splatting can only perform NVS by per-scene training

Introduction

Three Challenging Tasks

World Model (VR)

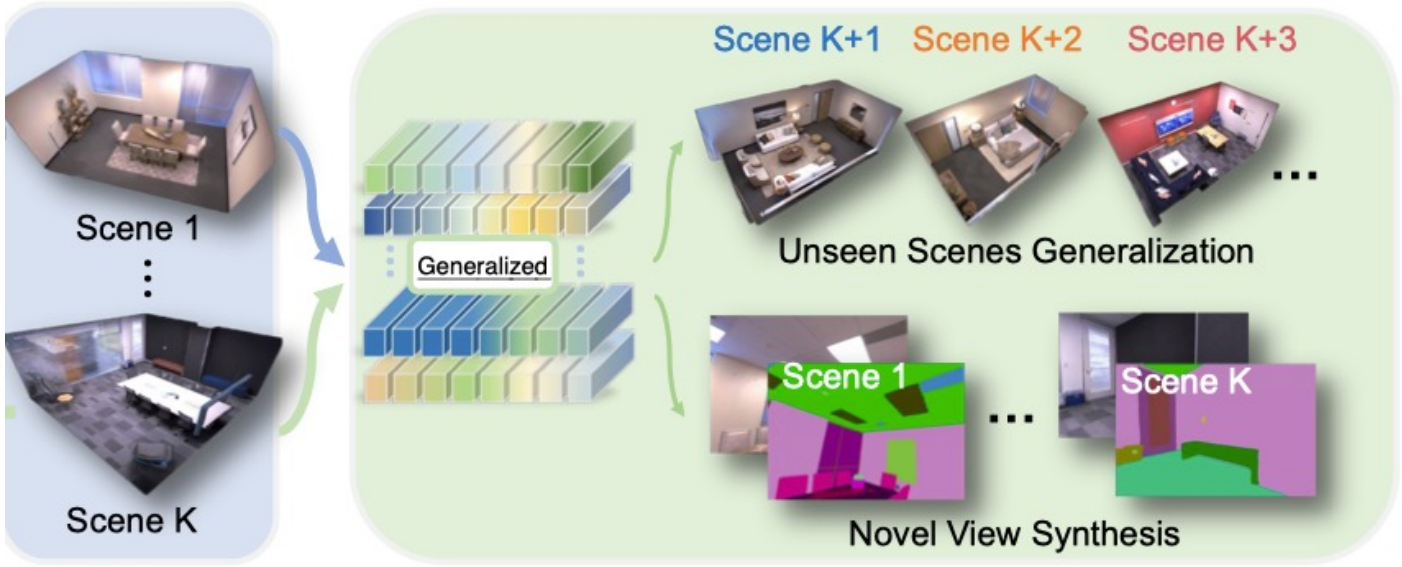
Street Views

Objects

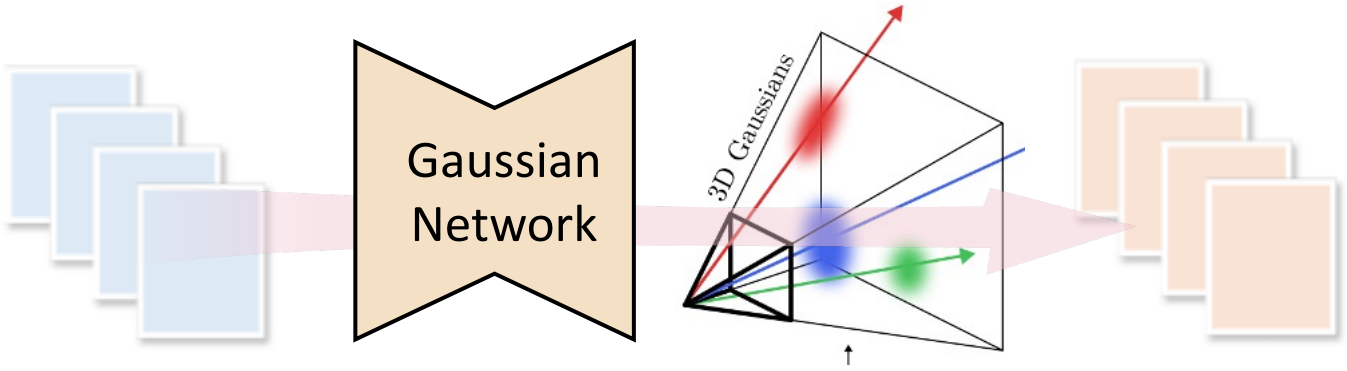
Per-scene optimized 3D-GS unable to handle such complex and large data

Introduction

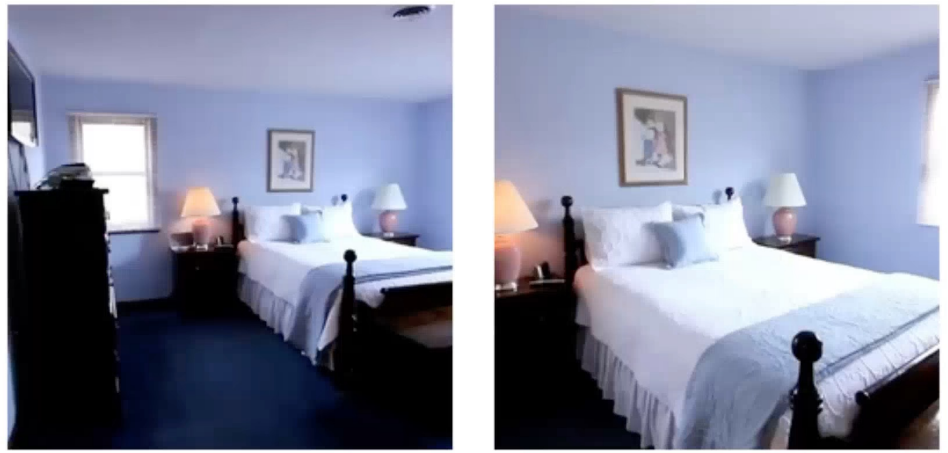
Introduction of Generalize Setting

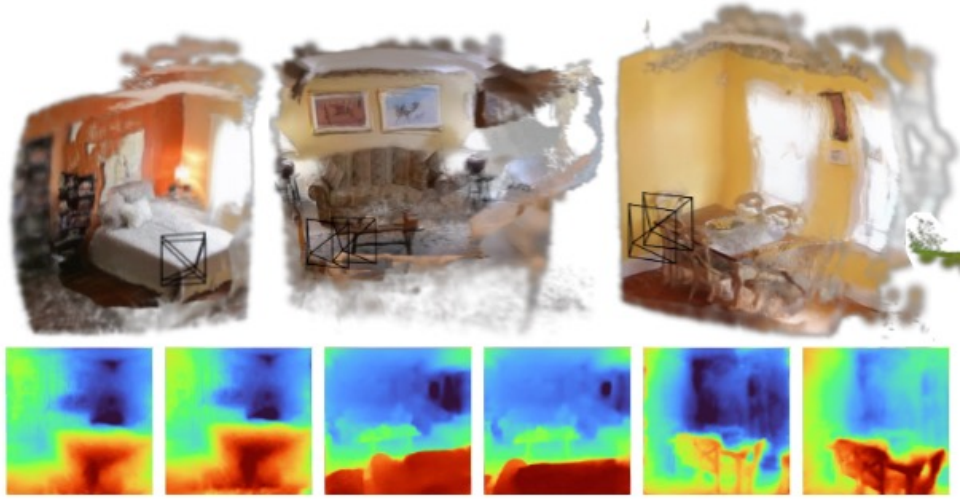


Generalizable 3D-GS enable to render novel-views on unseen scenes without training

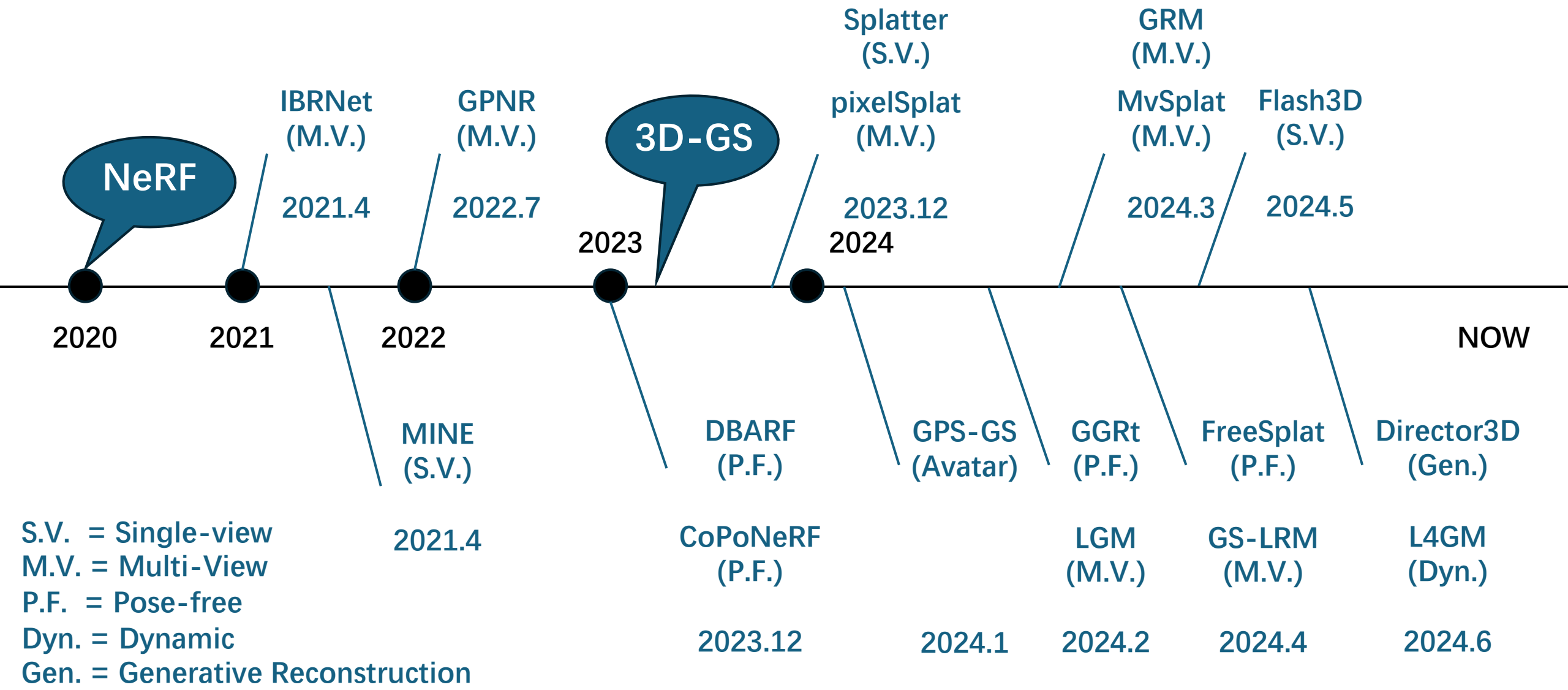


2 Input Views





Timeline of Generalizable NeRF / 3D-GS



S.V. = Single-view
M.V. = Multi-View
P.F. = Pose-free
Dyn. = Dynamic
Gen. = Generative Reconstruction

Topics

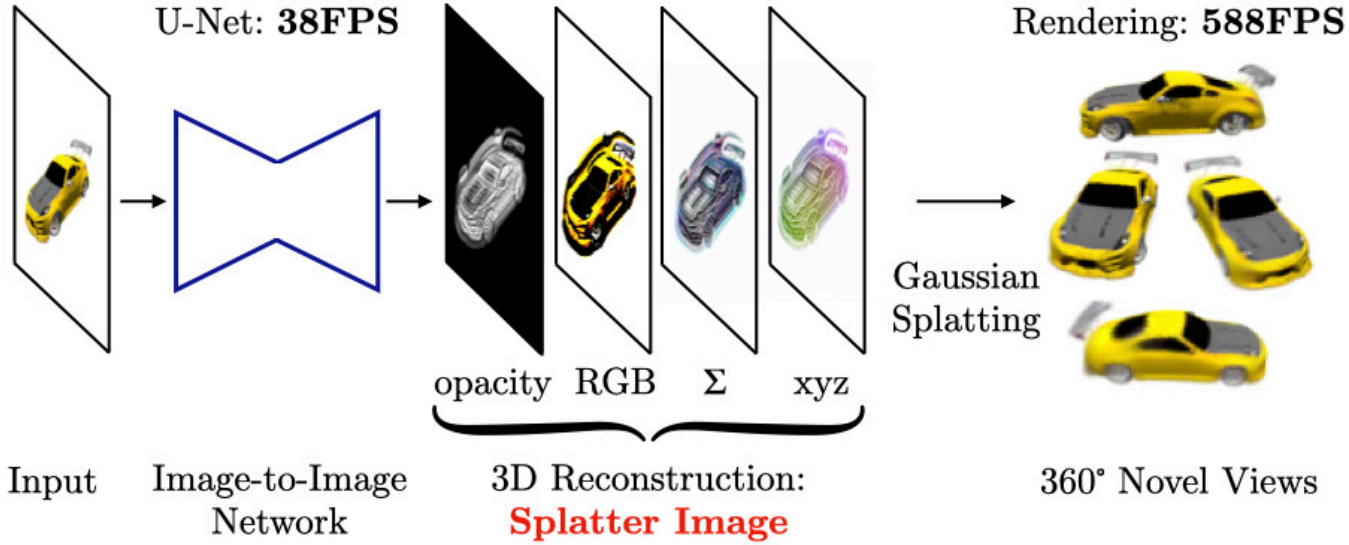
1. **Single-View** Generalizable Gaussian Model
2. **Multi-View** Generalizable Gaussian Model
3. **Generative** Generalizable Gaussian Model
4. **Pose-free** Generalizable Gaussian Model
5. Discussion of Future

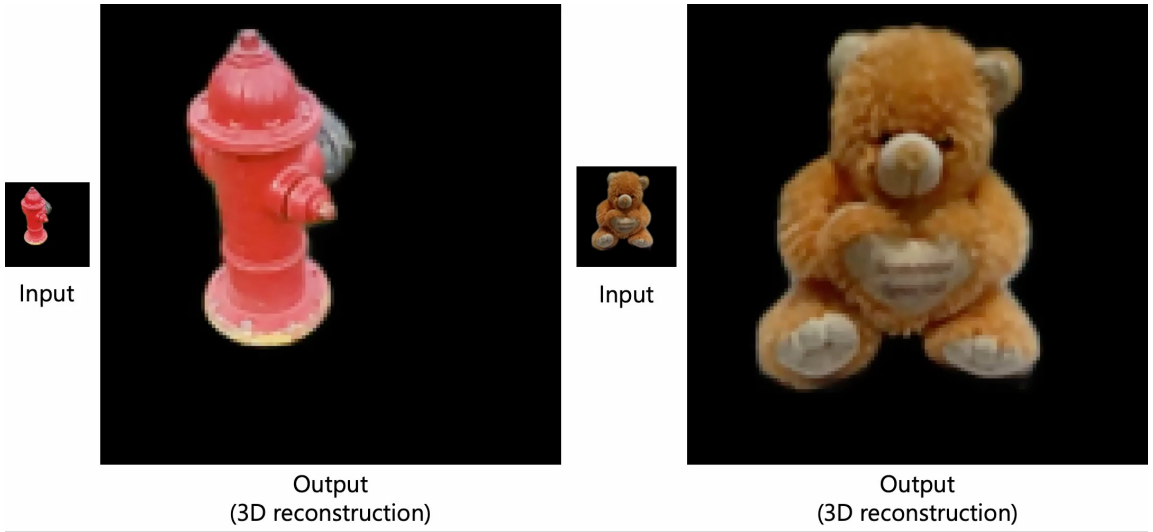
1. **Single-View** Generalizable Gaussian Model

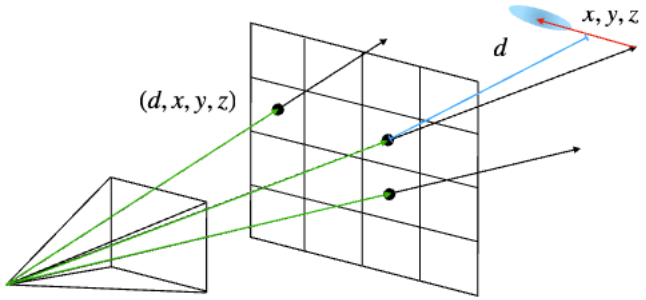
Single-View Generalizable 3D-GS

Splatter-Image, VGG Oxford, CVPR 2024

Splatter-Image uses an image-to-image neural network to map the input image to another image.







It holds the parameters of one coloured 3D Gaussian per pixel.

Figure 2. **Predicting locations.** The location of each Gaussian is parameterised by depth d and a 3D offset $\Delta = (\Delta_x, \Delta_y, \Delta_z)$. The 3D Gaussians are projected to depth d (blue) along camera rays (green) and moved by the 3D offset Δ (red).

$$\boldsymbol{\mu} = \begin{bmatrix} u_1 d + \Delta_x \\ u_2 d + \Delta_y \\ d + \Delta_z \end{bmatrix}$$

Single-View Generalizable 3D-GS

Splatter-Image, VGG Oxford, CVPR 2024

Performs poorly in the rear.

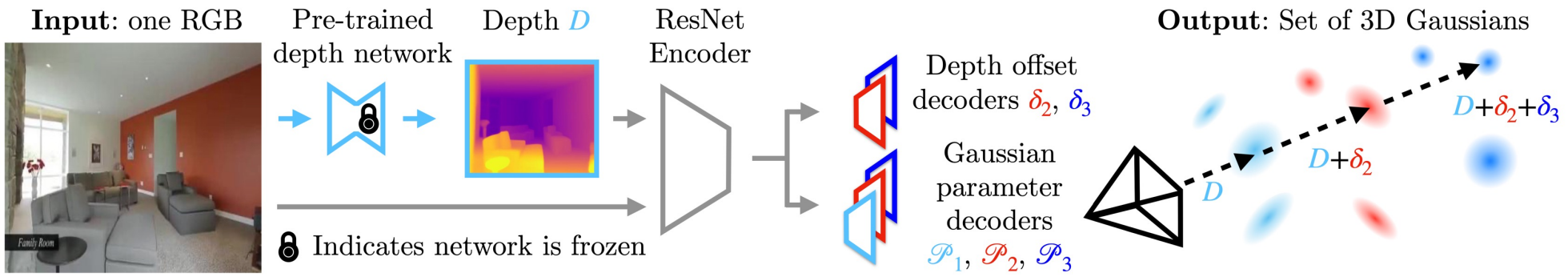
Splatter warps the Gaussians
from the front to the rear side

Splatter Image can only handle object-level environment.
It can’t handle unbounded issue (scale ambiguity)

Single-View Generalizable 3D-GS

Flash3D, VGG Oxford, Arxiv 2024

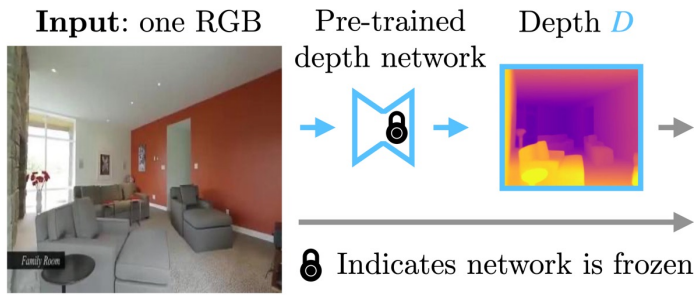
By introducing off-the-shelf Depth estimation method, **Flash3D** achieve 3D reconstruction in single view



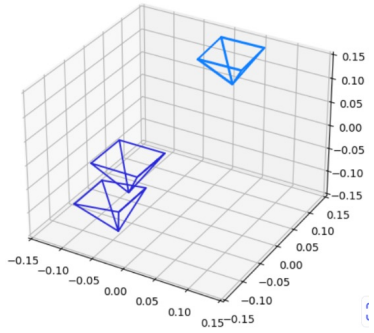
Input: 1 RGB image

Single-View Generalizable 3D-GS

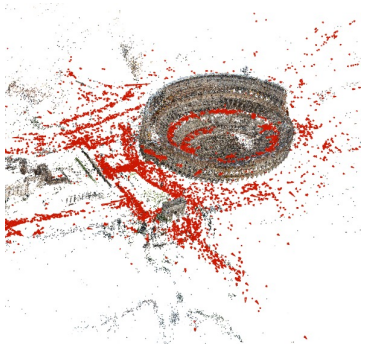
Flash3D, VGG Oxford, Arxiv 2024



It uses pre-trained depth model to directly predict XYZ of the Gaussians.



Single-view setting simplifies the scale ambiguity V.S. multi-view



Use COLMAP (sfm) to align the depth's scale and pose's scale

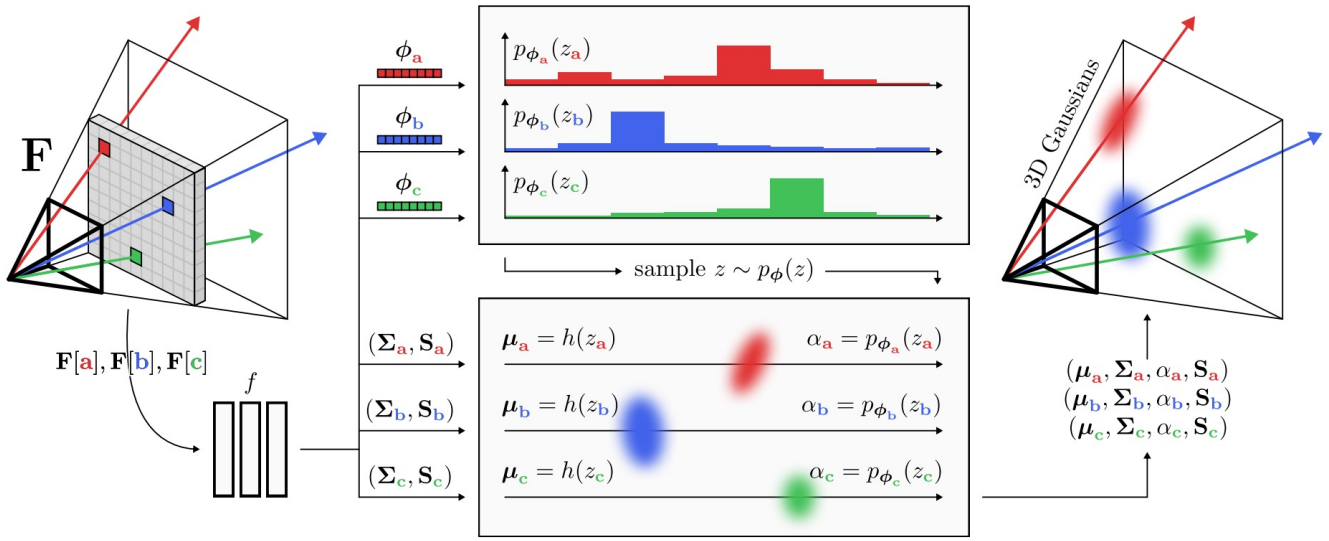
Flash3D still performs poorly on unseen regions

2. Multi-View Generalizable Gaussian Model

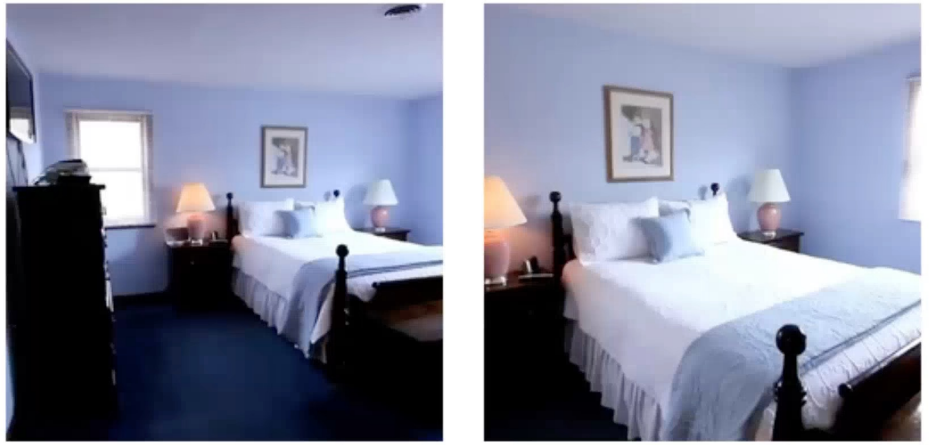
Multi-View Generalizable 3D-GS

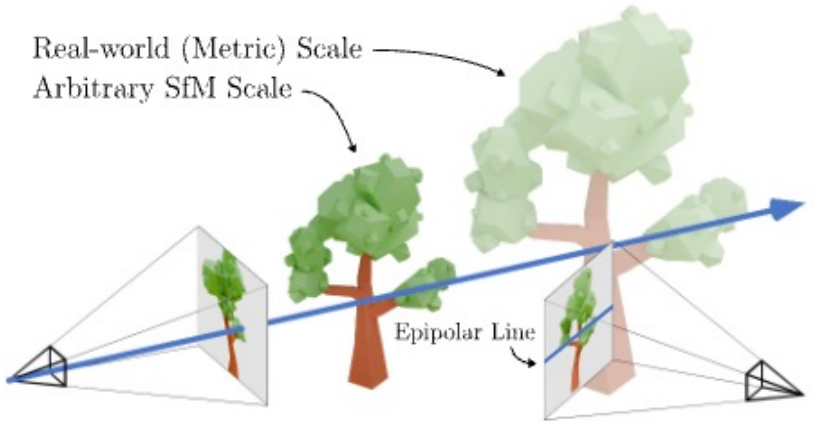
pixelSplat, MIT, CVPR 2024 Best Candidate

pixelSplat infers a 3D Gaussian scene from two input views in a single forward pass.



2 Input Views





It uses epipolar encoder (cross-attention / self-attention) to handle the scale problem, like previous NeRF (i.e. GPNR, GNT)

$$\mathbf{s} = \tilde{\mathbf{F}}[\tilde{\mathbf{u}}_l] \oplus \gamma(\tilde{d}_{\tilde{\mathbf{u}}_l}) \quad (1)$$

$$\mathbf{q} = \mathbf{Q} \cdot \mathbf{F}[\mathbf{u}], \quad \mathbf{k}_l = \mathbf{K} \cdot \mathbf{s}, \quad \mathbf{v}_l = \mathbf{V} \cdot \mathbf{s}, \quad (2)$$

$$\mathbf{F}[\mathbf{u}] \mathrel{+}= \mathrm{Att}(\mathbf{q}, \{\mathbf{k}_l\}, \{\mathbf{v}_l\}), \quad (3)$$

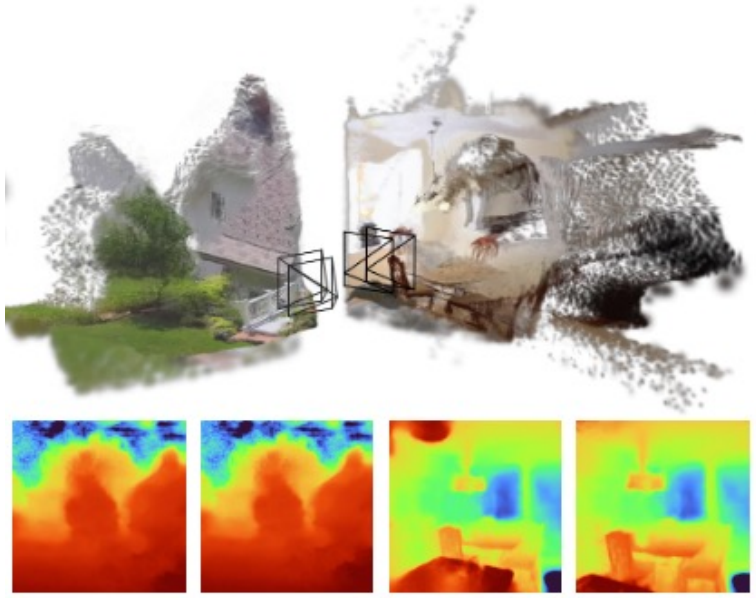
Multi-View Generalizable 3D-GS

pixelSplat, MIT, CVPR 2024 Best Candidate

Memory Consumption

Ref. View	Resolution	PSNR ↑ Gen.	PSNR ↑ Ft	Mem. (GB) Gen.	Mem. (GB) Ft
2	384 × 496	27.65	27.77	28.48	29.03
3	384 × 496	-	-	OOM	OOM
5	192 × 248	26.00	28.51	31.05	31.05

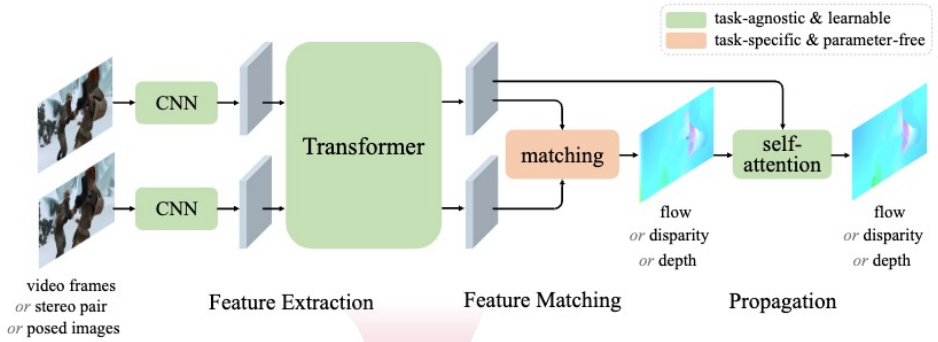
Rough Geometry Representation



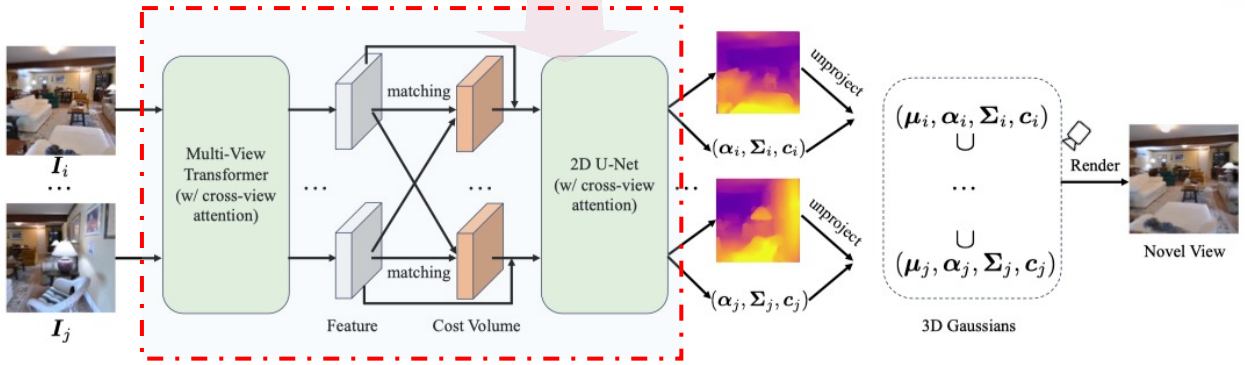
Multi-View Generalizable 3D-GS

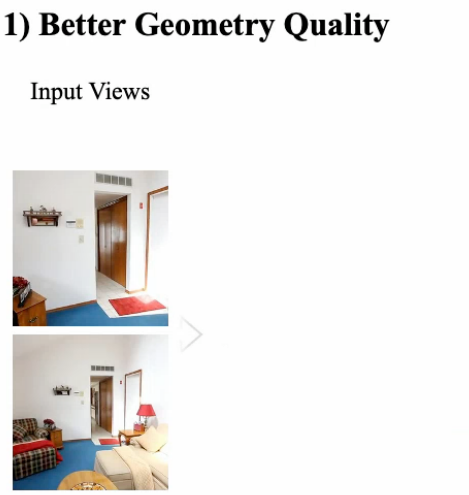
MVSplat, CVG ETH, ECCV 2024

MVSplat builds a cost volume representation to efficiently predict 3D Gaussians from sparse multi-view images in a single forward pass.



Unify Flow, Stereo and Depth Estimation, TPAMI'23





insight: Gaussians prediction has strong connection with depth estimation

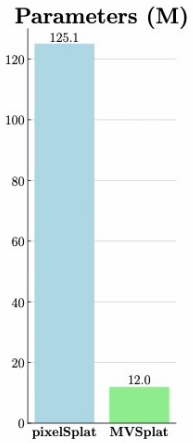
Multi-View Generalizable 3D-GS

MVSplat, CVG ETH, ECCV 2024

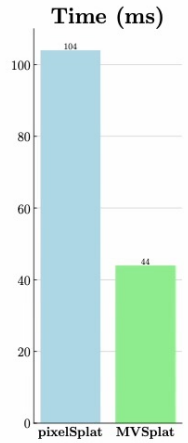
Simple but efficiency solutions

Compared to the latest SOTA pixelSplat, our **MVSplat** has

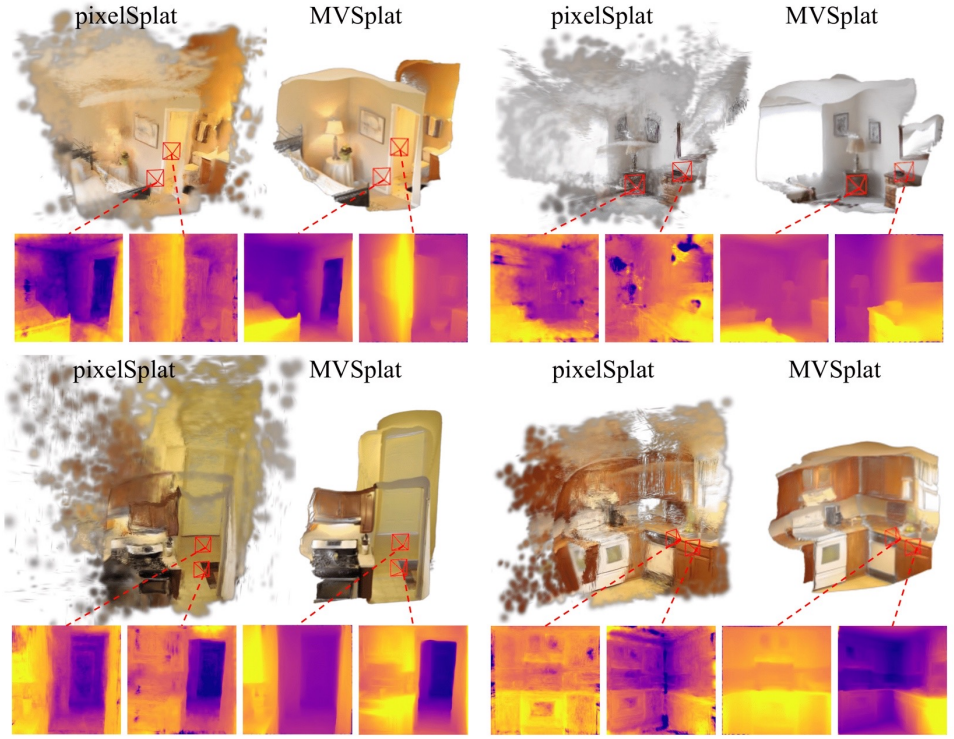
> 10× Fewer Parameters



> 2× Faster Running Speed

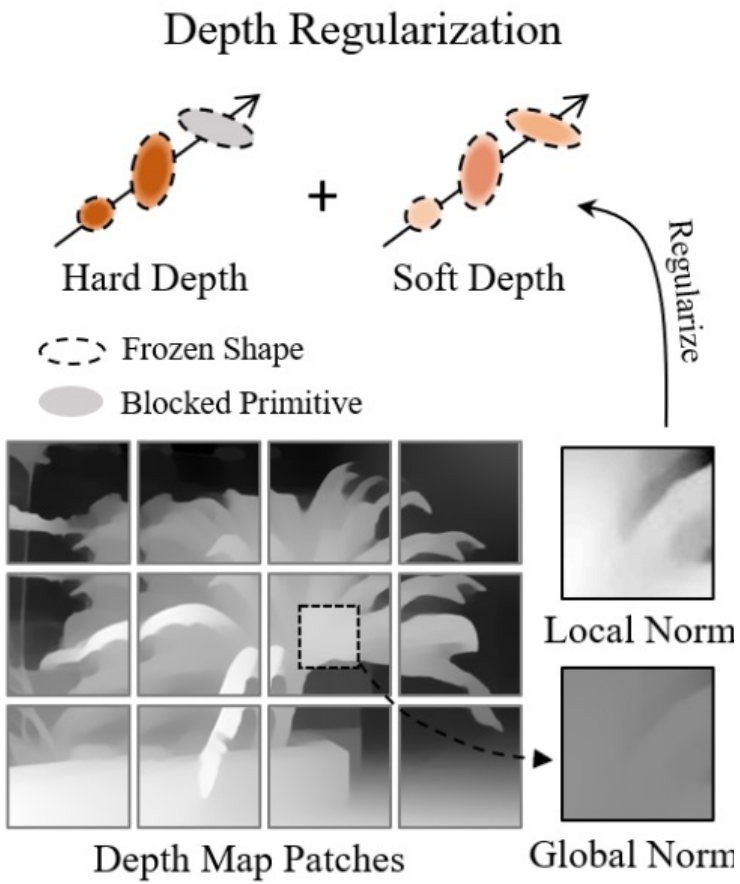


Precise Geometry Representation



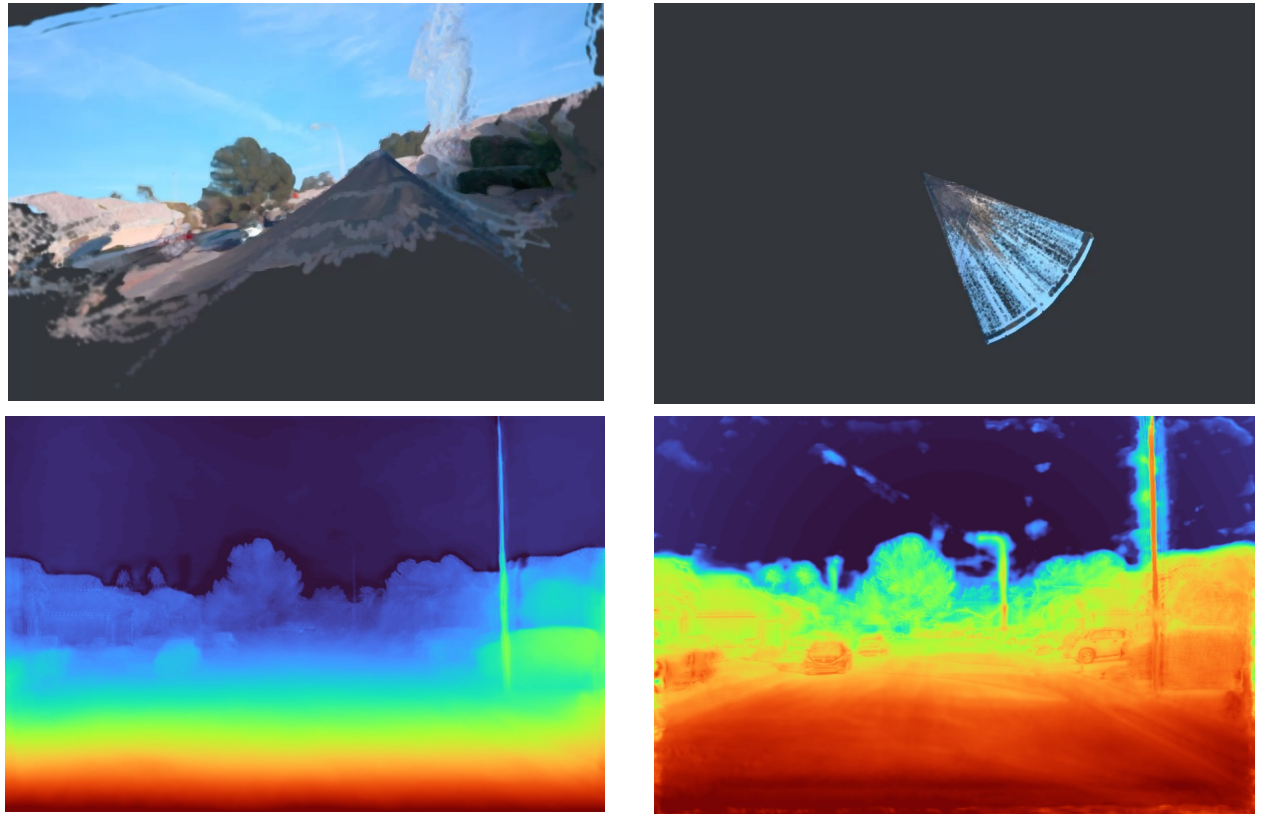
Multi-View Generalizable 3D-GS

MVSplat, CVG ETH, ECCV 2024



w/ Depth Opt.

w/o Depth Opt.



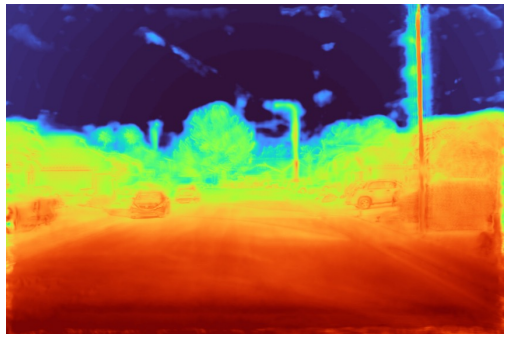
Geo-Representation can be **further optimized** by loss supervision

Multi-View Generalizable 3D-GS

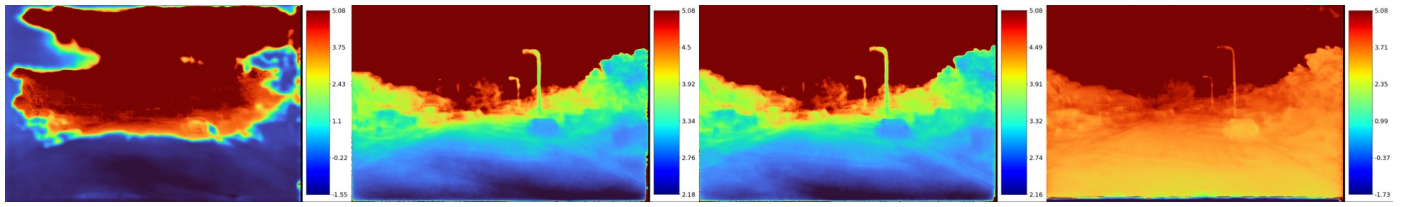
MVSplat, CVG ETH, ECCV 2024

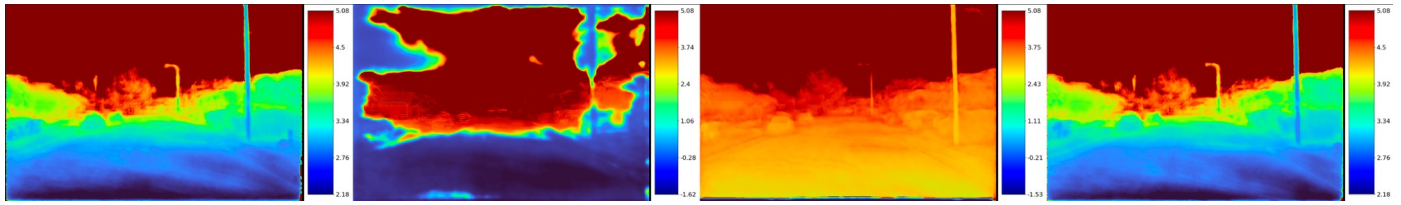
Rendered Image

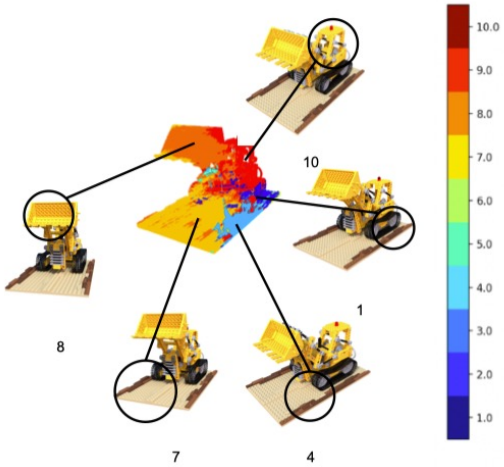
Rendered Depth



Rendered by input views







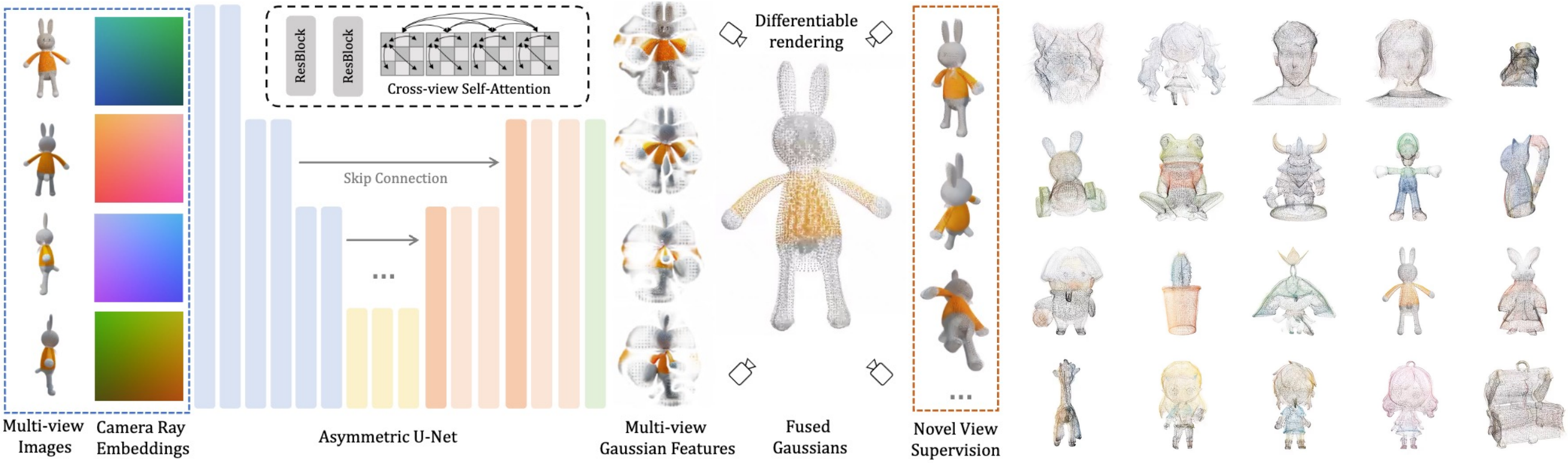
Contributions of input views for final rendered view, by GNT

New finding: similar with generalizable NeRF (GNT), the rendered image is assembled by input views. However, their depths are also reflected by the rendered supervision, which should be consistent.

Question: should we learn more from previous depth estimation networks?

Multi-View Generalizable 3D-GS

LGM, MMLab NTU, ECCV 2024

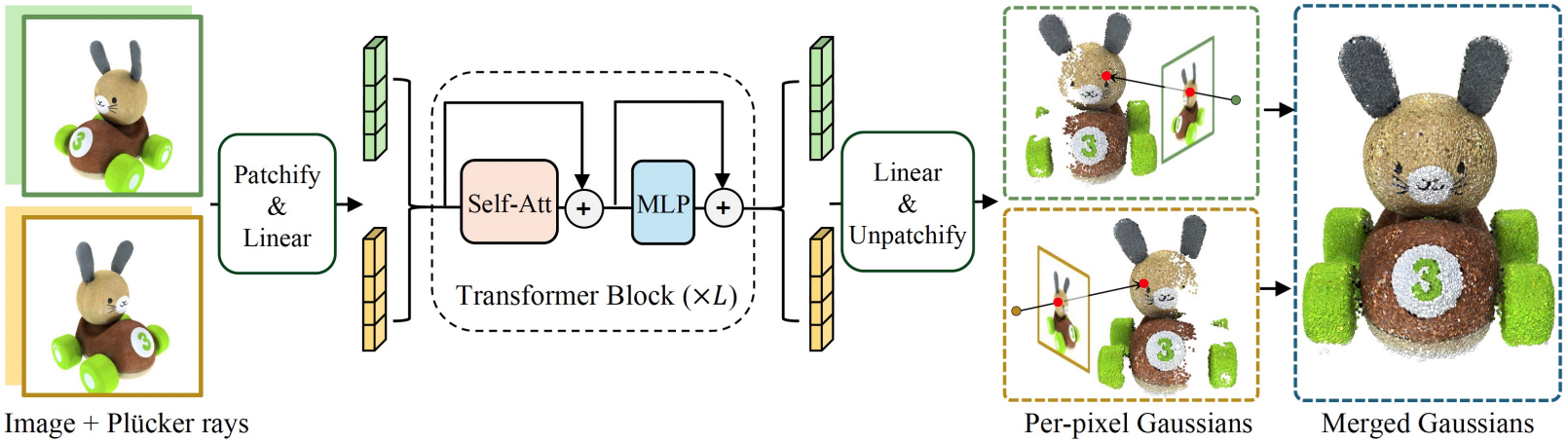


32 NVIDIA A100 (80G) GPUs for about 4 days

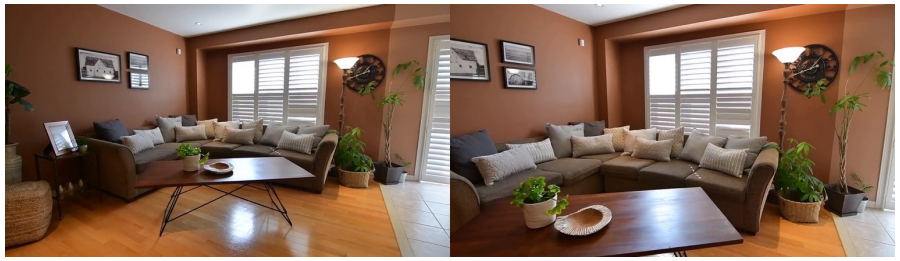
Large model wins thousands tricks

Multi-View Generalizable 3D-GS

GS-LRM, Adobe, ECCV 2024



Input views



Novel views

GS-LRM achieve significant improvement compared with LGM and PixelSplat

	GSO			ABO		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Triplane-LRM [32]	26.54	0.893	0.064	27.50	0.896	0.093
Ours (Res-512)	**30.52**	**0.952**	**0.050**	**29.09**	**0.925**	**0.085**
LGM [61]	21.44	0.832	0.122	20.79	0.813	0.158
Ours (Res-256)	**29.59**	**0.944**	**0.051**	**28.98**	**0.926**	**0.074**

	RealEstate10k		
	PSNR ↑	SSIM ↑	LPIPS ↓
pixelNeRF [72]	20.43	0.589	0.550
GPNR [57]	24.11	0.793	0.255
Du et. al [22]	24.78	0.820	0.213
pixelSplat [8]	25.89	0.858	0.142
Ours	**28.10**	**0.892**	**0.114**

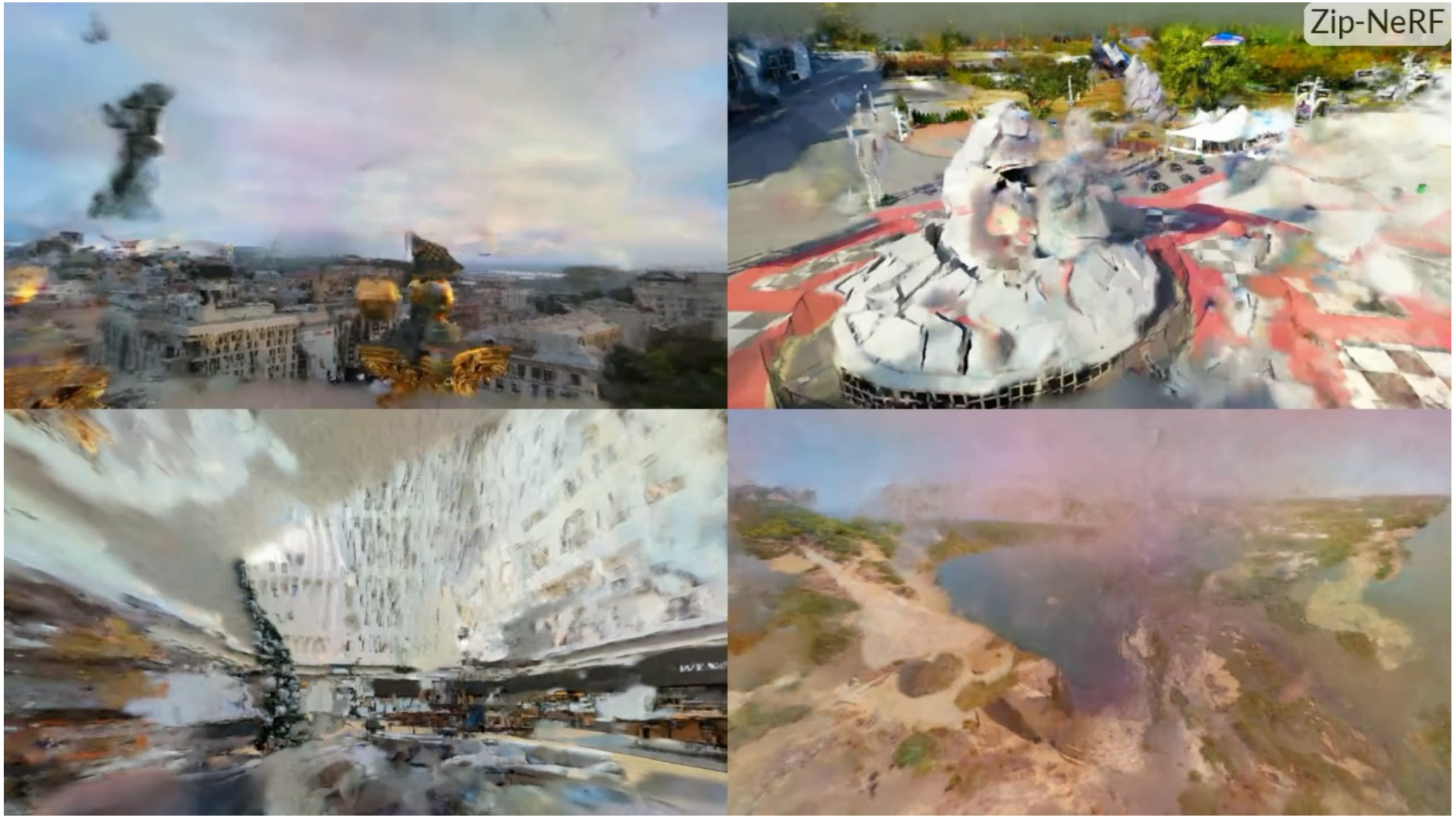
Use 64 A100 (40G VRAM) GPUs with 2 days training

Large model wins thousands tricks

3. Generative Generalizable Gaussian Model

Generative Generalizable 3D-GS

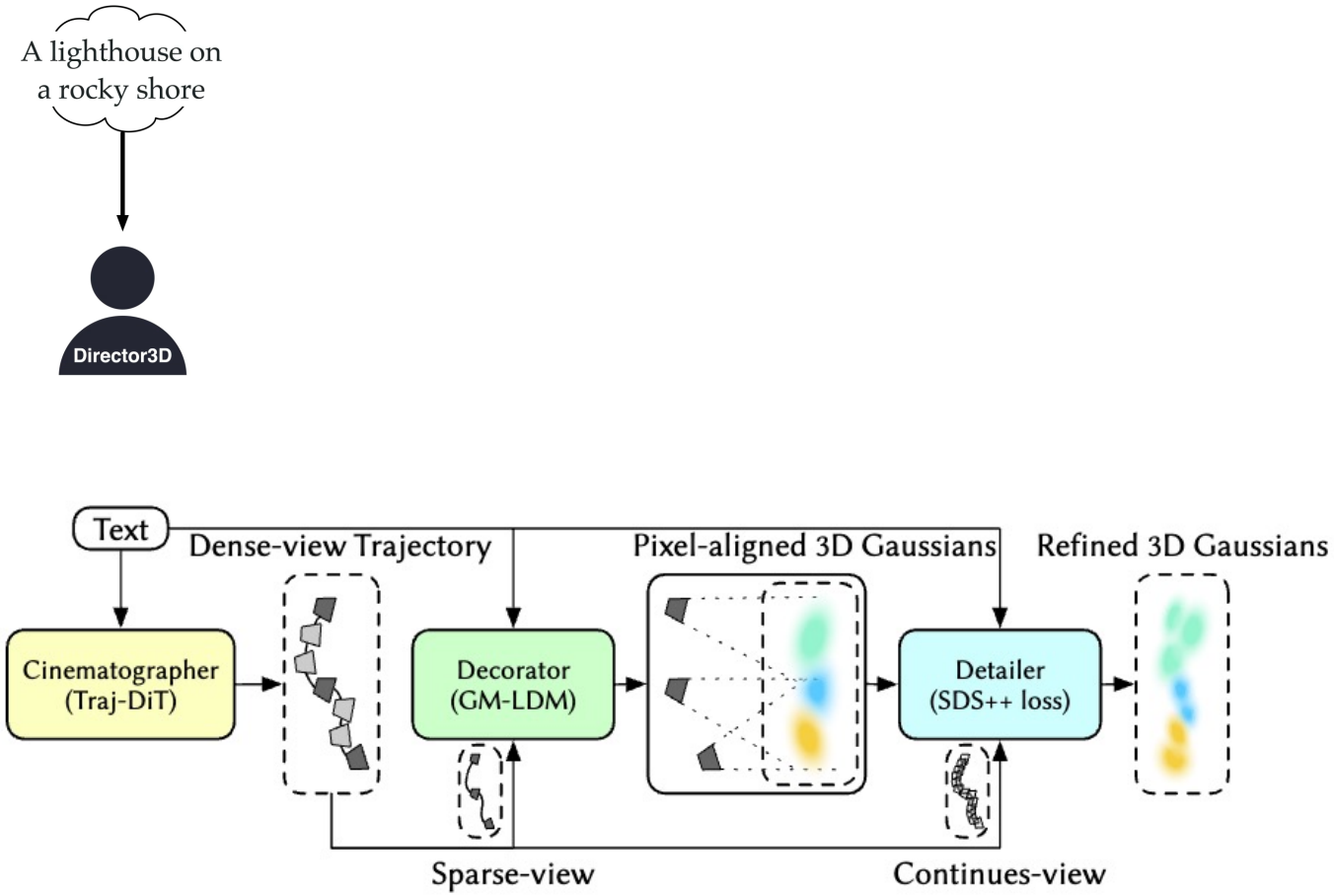
Director3D, Shanghai AI Lab, Arxiv 2024



Traditional NeRF v.s. NeRF + Diffusion

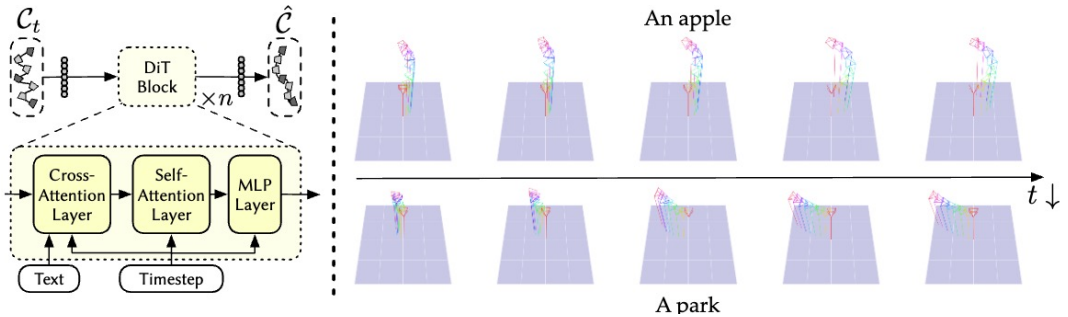
Generative Generalizable 3D-GS

Director3D, Shanghai AI Lab, Arxiv 2024

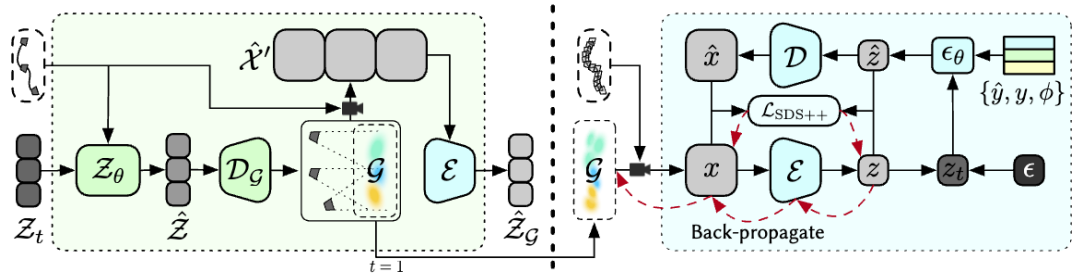


Main framework of **Director3D**.

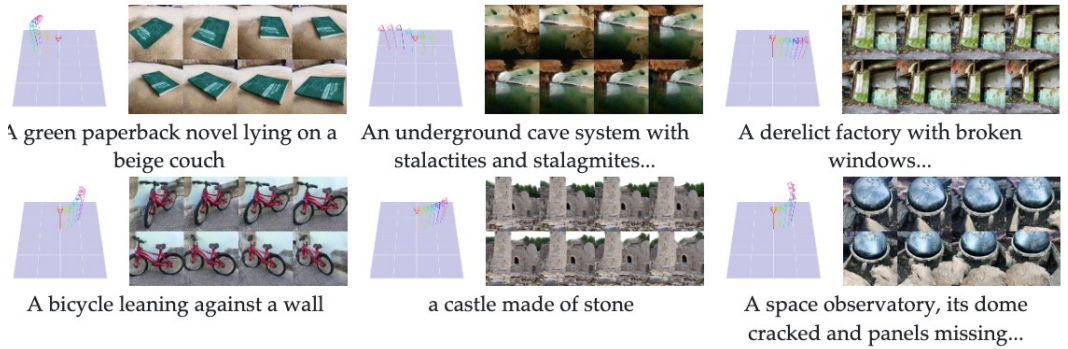
Architecture of Traj-DiT and some DEMO



Architecture of GM-LDM



Demo

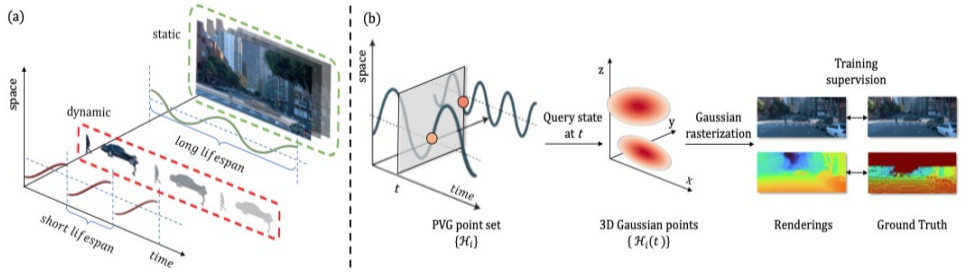


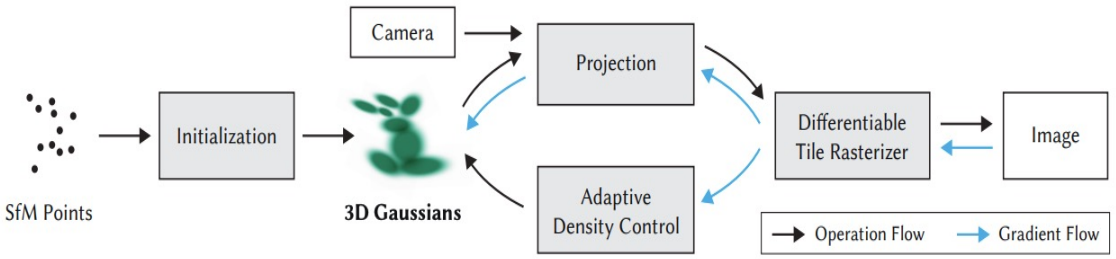
4. **Pose-free** Generalizable Gaussian Model

Pose-free Generalizable 3D-GS

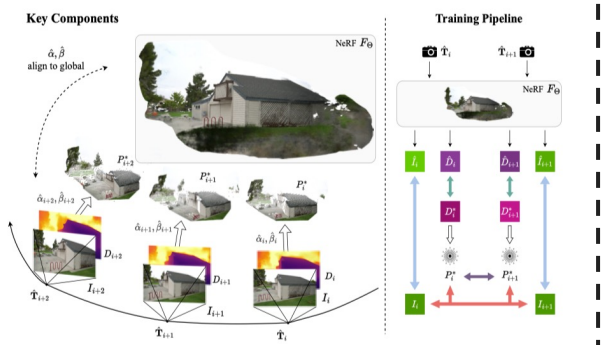
GGRt, NWPU / Baidu, ECCV 2024

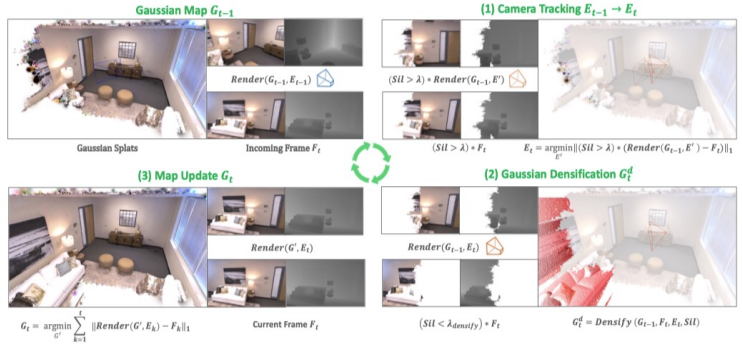
Current 3D-GS based methods need precise poses prior to conduct novel-view rendering tasks

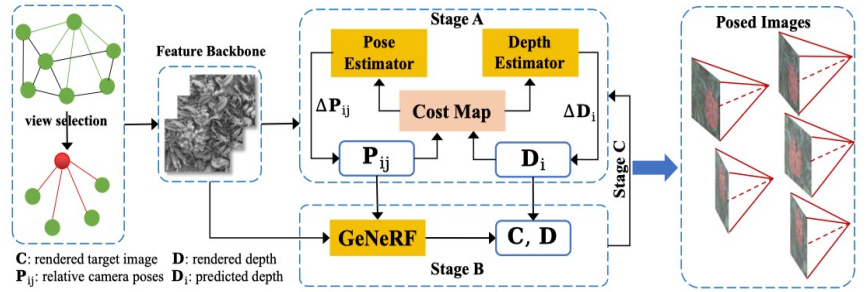


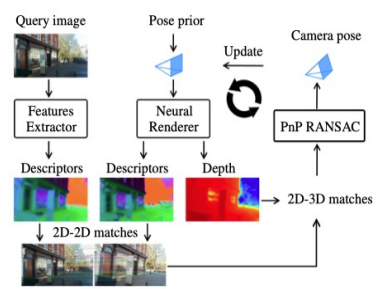


Existing Pose-free NeRF methods require massive time to train single scene while Pose-free 3D-GS lacks generalization ability



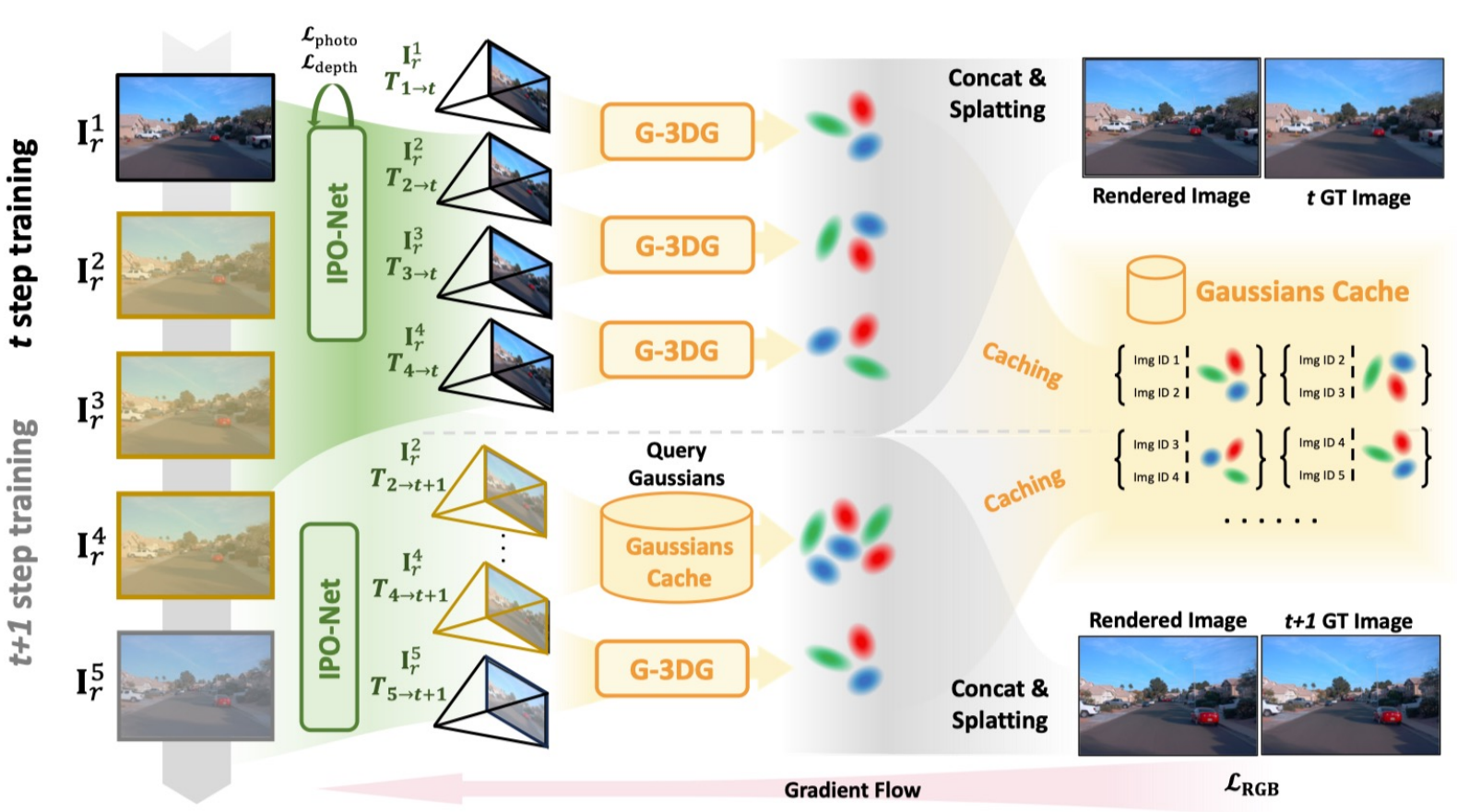


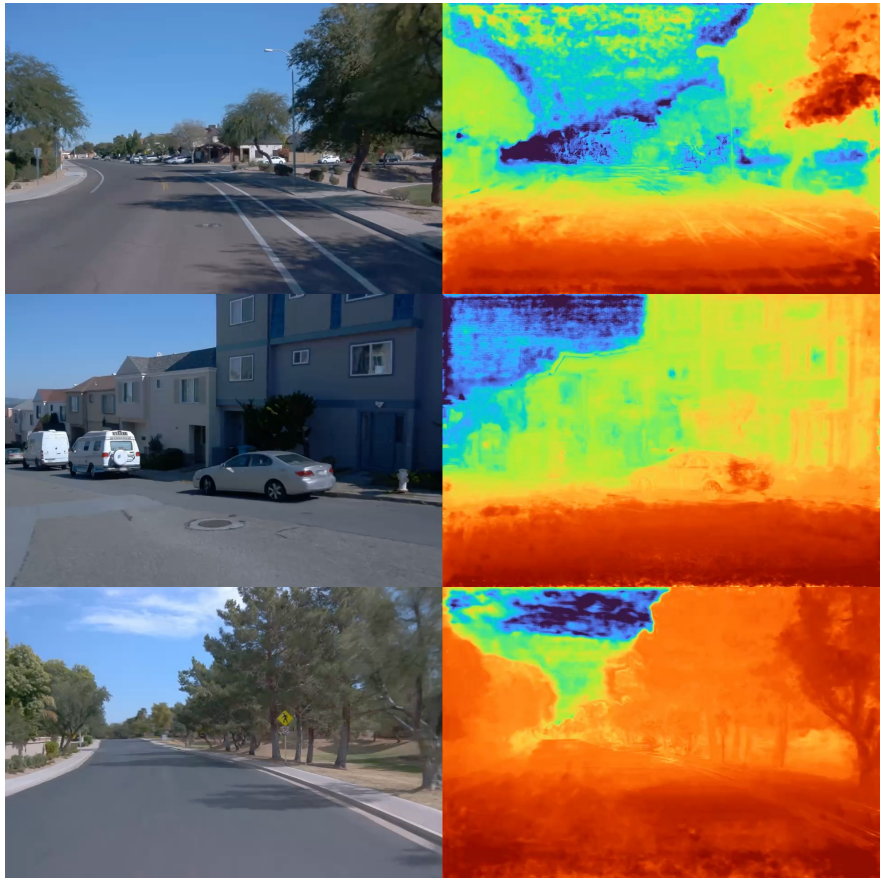




Pose-free Generalizable 3D-GS

GGRt, NWPU / Baidu, ECCV 2024

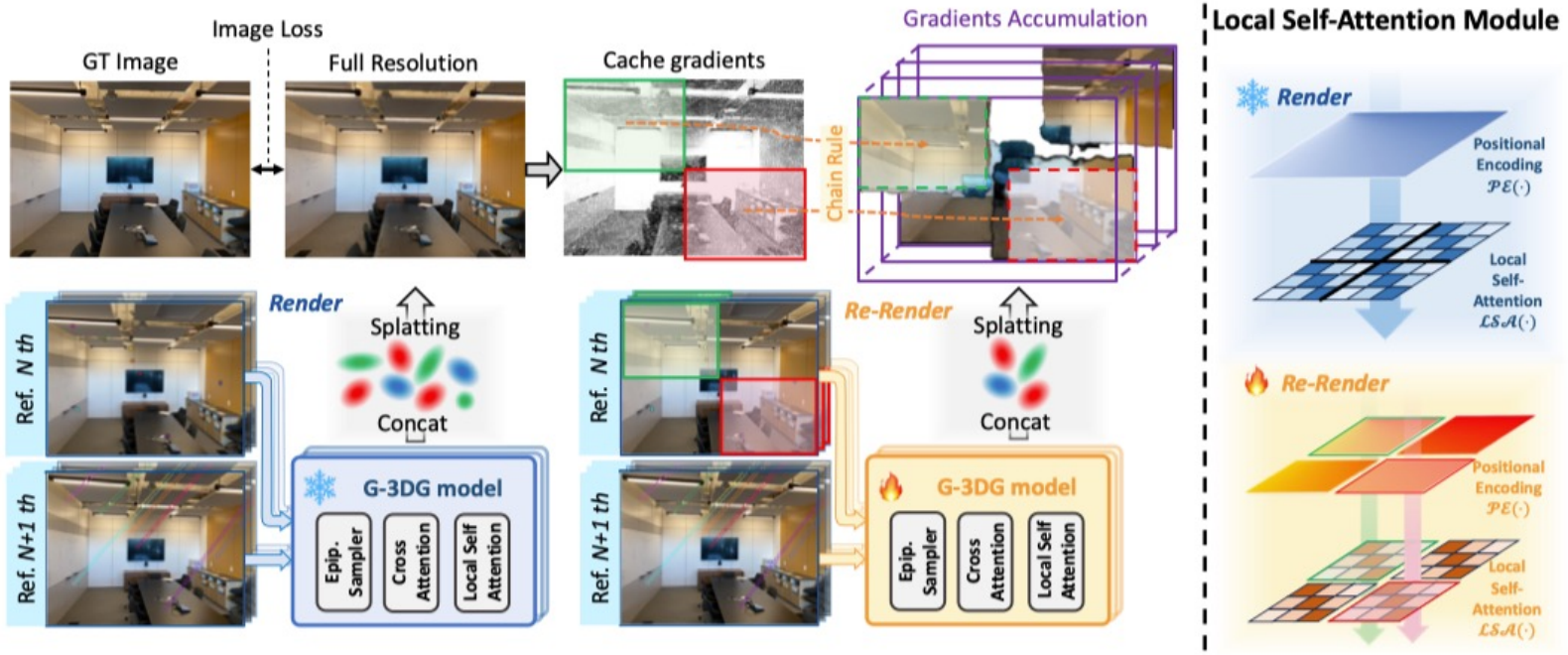


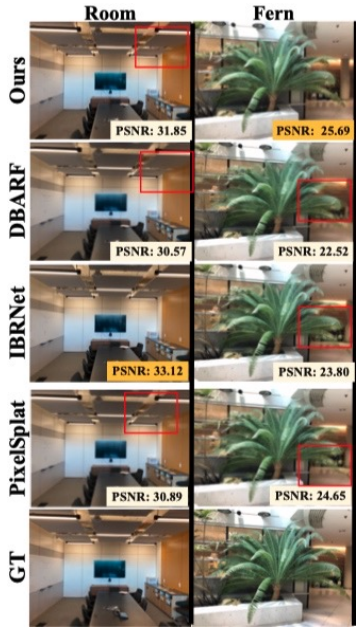


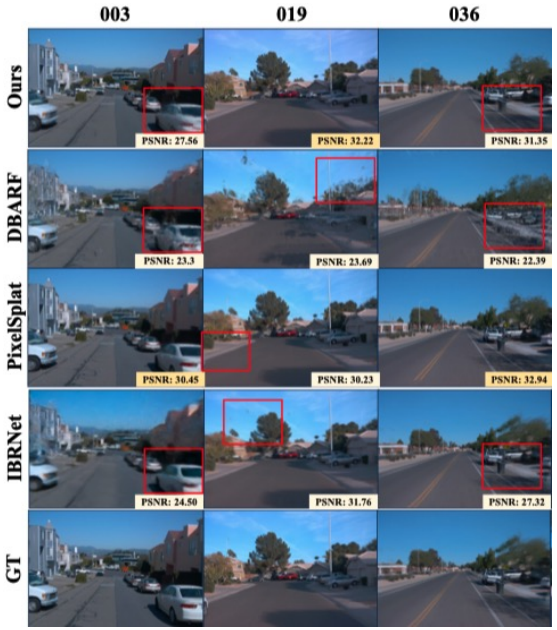
The first pose-free generalizable 3D-GS framework, **GGRt** achieves inference at ≥ 5 FPS and real-time rendering at ≥ 100 FPS.

Pose-free Generalizable 3D-GS

GGRt, NWPU / Baidu, ECCV 2024





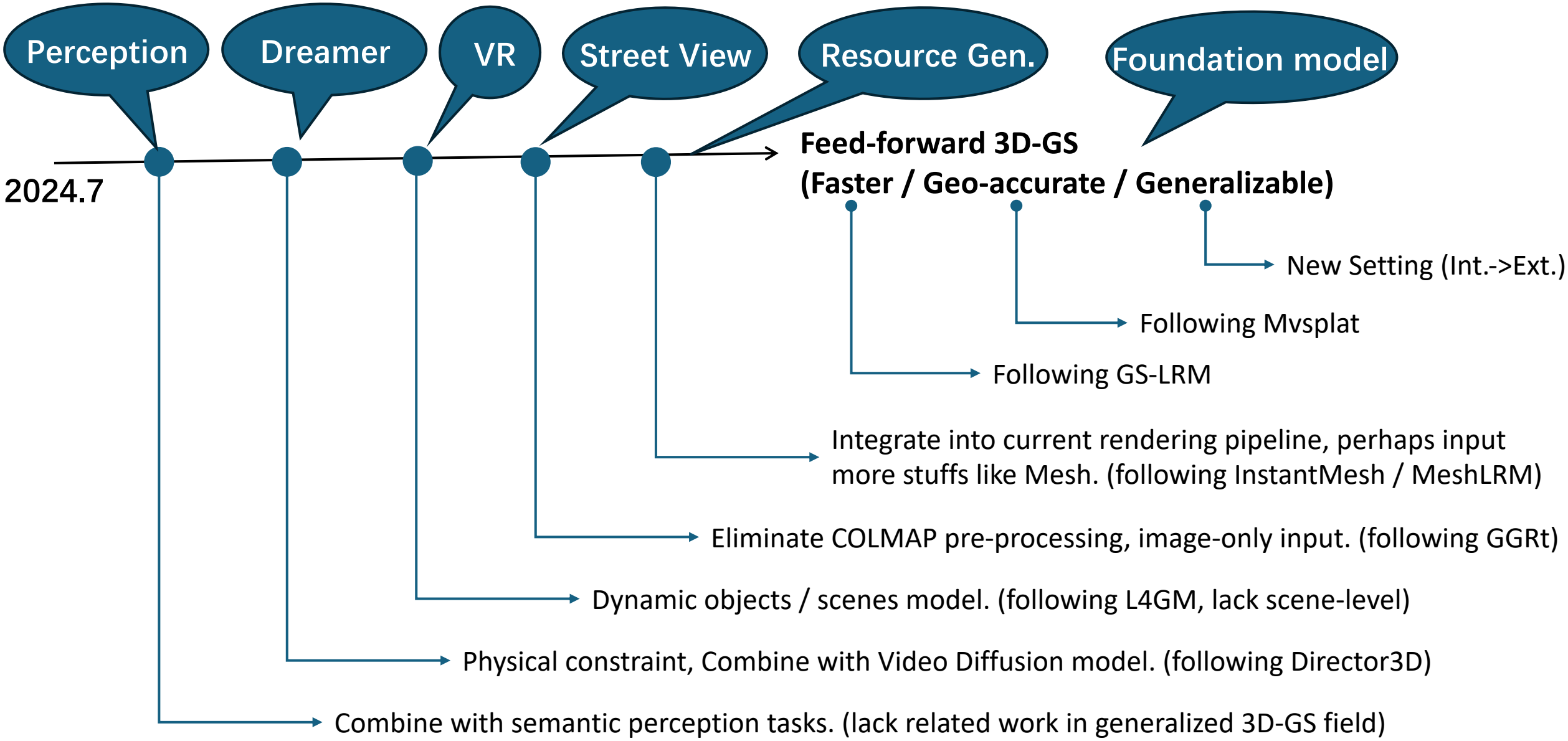


Deferred back-propagation mechanism that enables high-resolution training and inference, overcoming the resolution constraints of previous methods.

Method	Ref. View	Resolution	PSNR ↑ Gen.	PSNR ↑ Ft	Mem. (GB) Gen.	Mem. (GB) Ft
Our *w/o* Defer	2	384 × 496	27.65	27.77	28.48	29.03
Our *w/o* Defer	3	384 × 496	-	-	OOM	OOM
Our *w/o* Defer	5	192 × 248	26.00	28.51	31.05	31.05
Our *w/* Defer	5	384 × 496	29.02	29.85	29.35	29.35
Our *w/* Defer	7	384 × 496	-	31.50	-	34.61

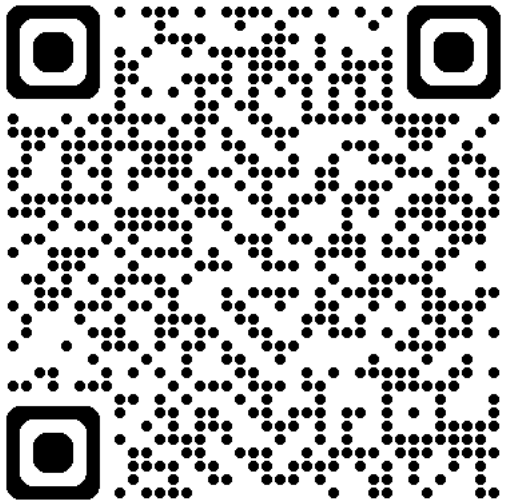
5. **Discussion** of Future

Pathway

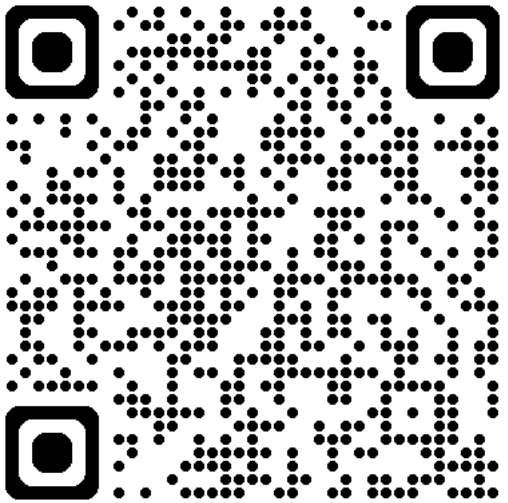

Perception
Dreamer
VR
Street View
Resource Gen.
Foundation model
2024.7
Feed-forward 3D-GS
(Faster / Geo-accurate / Generalizable)
New Setting (Int.->Ext.)
Following Mvsplat
Following GS-LRM
Integrate into current rendering pipeline, perhaps input more stuffs like Mesh. (following InstantMesh / MeshLRM)
Eliminate COLMAP pre-processing, image-only input. (following GGRt)
Dynamic objects / scenes model. (following L4GM, lack scene-level)
Physical constraint, Combine with Video Diffusion model. (following Director3D)
Combine with semantic perception tasks. (lack related work in generalized 3D-GS field)

Q&A

Thanks the invitations of GAMES Webinar!



Author Page



URL of PDF