



浙江大學
ZHEJIANG UNIVERSITY

ByteDance



MONASH
University



NEURAL INFORMATION
PROCESSING SYSTEMS

ZPressor: Bottleneck-Aware Compression for Scalable Feed-Forward 3DGs



Weijie Wang¹



Donny Y. Chen²



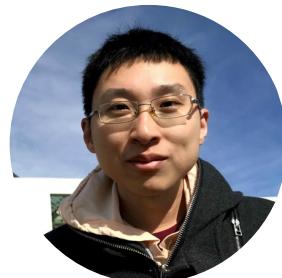
Zeyu Zhang³



Duochao Shi¹



Akide Liu³



Bohan Zhuang¹

¹Zhejiang University ²ByteDance Seed ³Monash University

Weijie Wang (王伟杰)

College of Computer Science and Technology, Zhejiang University

2025/11/6

About Me

»» Weijie Wang (王伟杰)

First-year Ph.D. student @ **ZIP Lab, State Key Lab of CAD&CG, ZJU**

Supervised by **Prof. Bohan Zhuang**

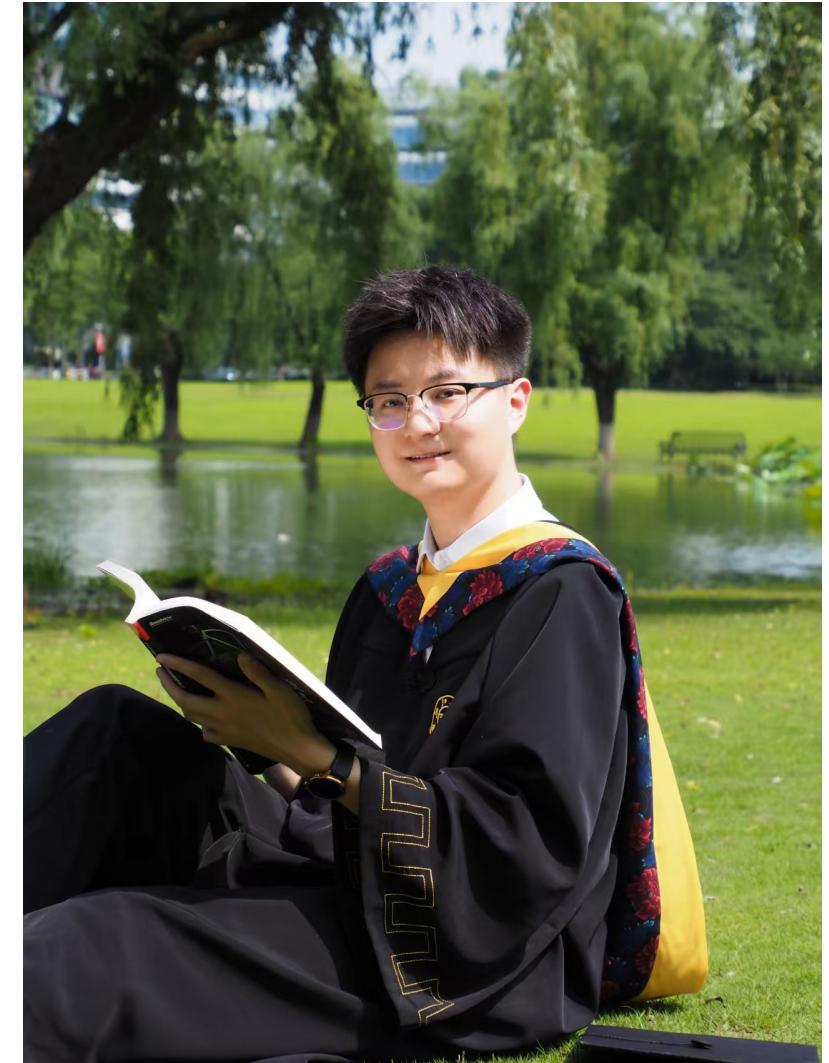
Research Interest:

- **Feed-Forward Reconstruction:** [ZPressor](#), [PM-Loss](#), [VolSplat](#)
- **Dynamic Reconstruction:** [Street Gaussians](#), [DriveGen3D](#)
- **Interactive Generation:** [WonderTurbo](#)

Webpage: <https://lhmd.top>

Email: wangweijie@zju.edu.cn

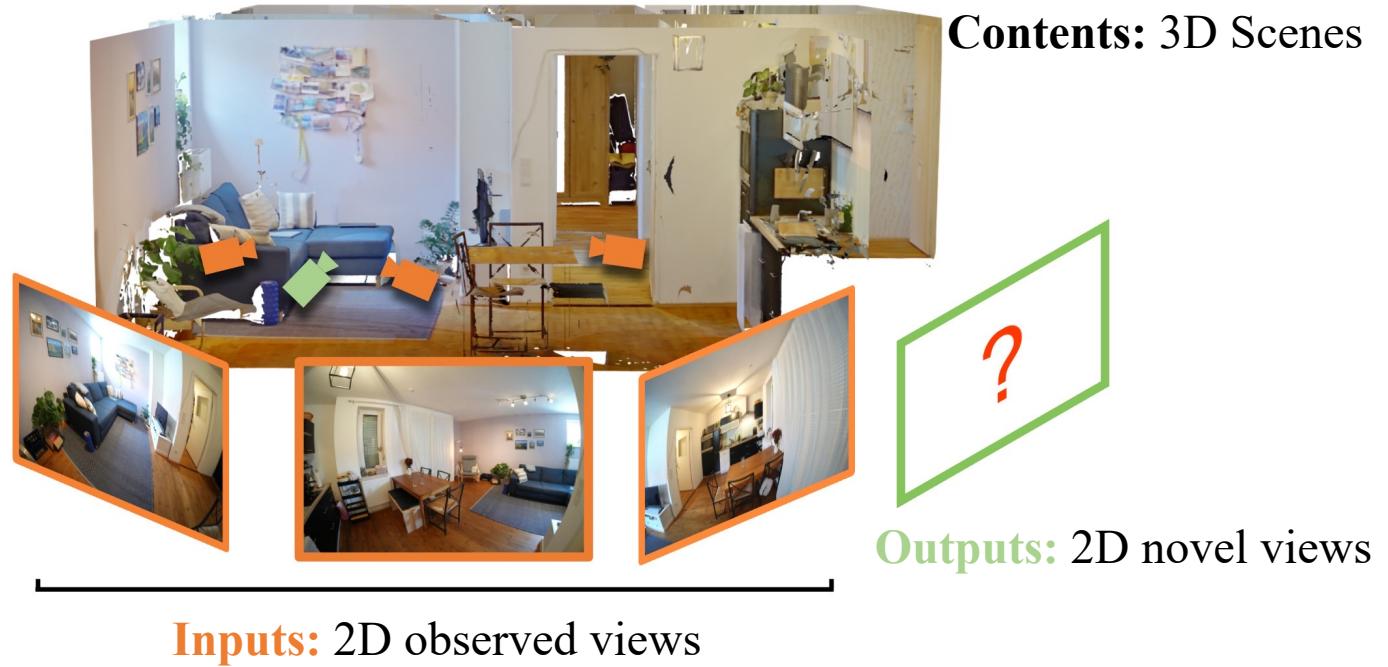
Wechat: zju-lhmd



Tasks

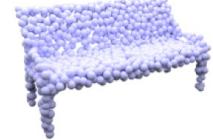
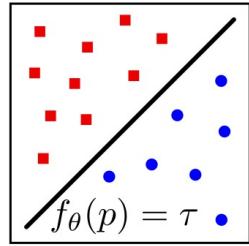
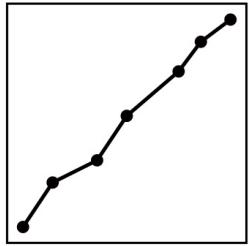
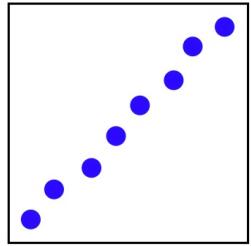
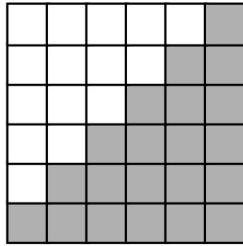


3D Reconstruction



Novel View Synthesis

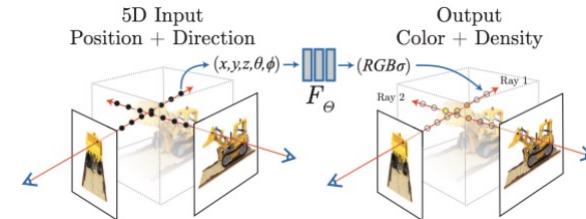
3D Representations



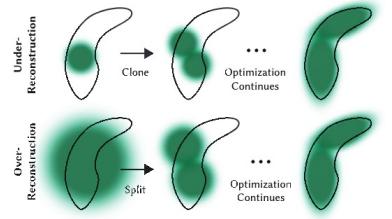
Voxel

Point Cloud

Mesh

Occupancy
Networks

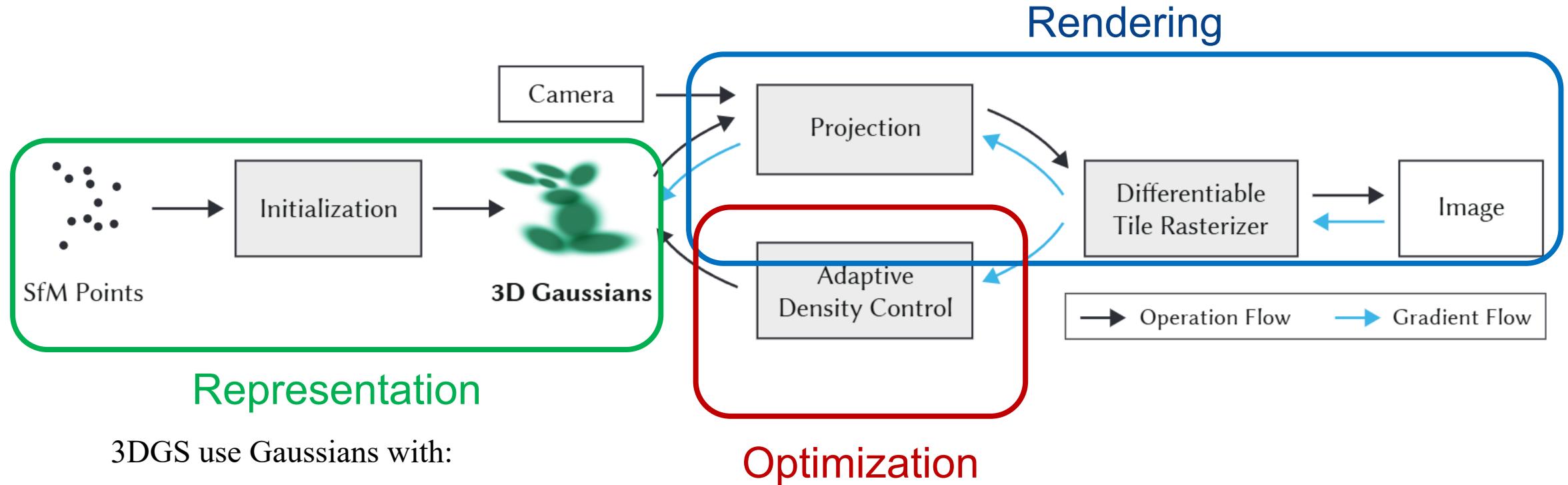
Neural Radiance Field (NeRF)



3D Gaussian Splatting (3DGS)

There is no canonical representation in 3D. We chose 3DGS since it performs the best for NVS in general.

3D Gaussian Splatting (3DGS)



3DGS use Gaussians with:

- μ : Gaussian center position (xyz)
- α : opacity; (how transparent)
- Σ : covariance; (scale, rotation)
- c : color; (spherical harmonic)

Limitations of Per-Scene based 3DGS

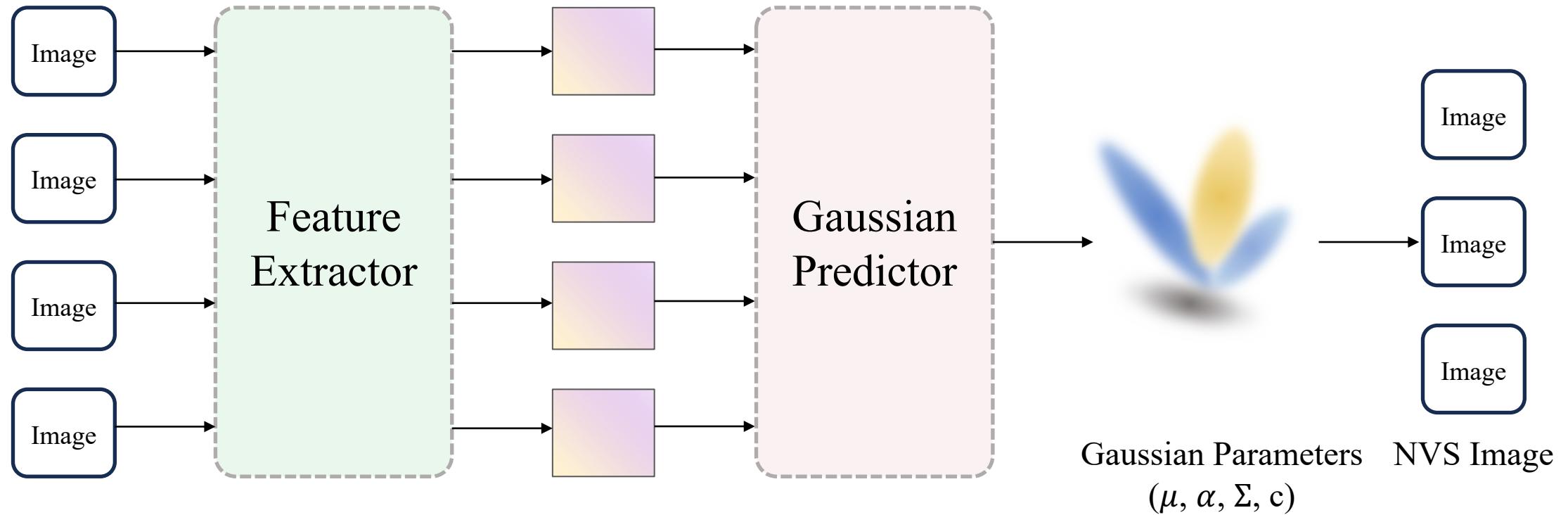
1. **Time:** requires applying the optimization process to *each scene* (20+ mins)
2. **Space:** requires additional permanent storage for the 3D representation of *each scene* (10+ M)



The bicycle scene takes: ~50 mins, ~100 M

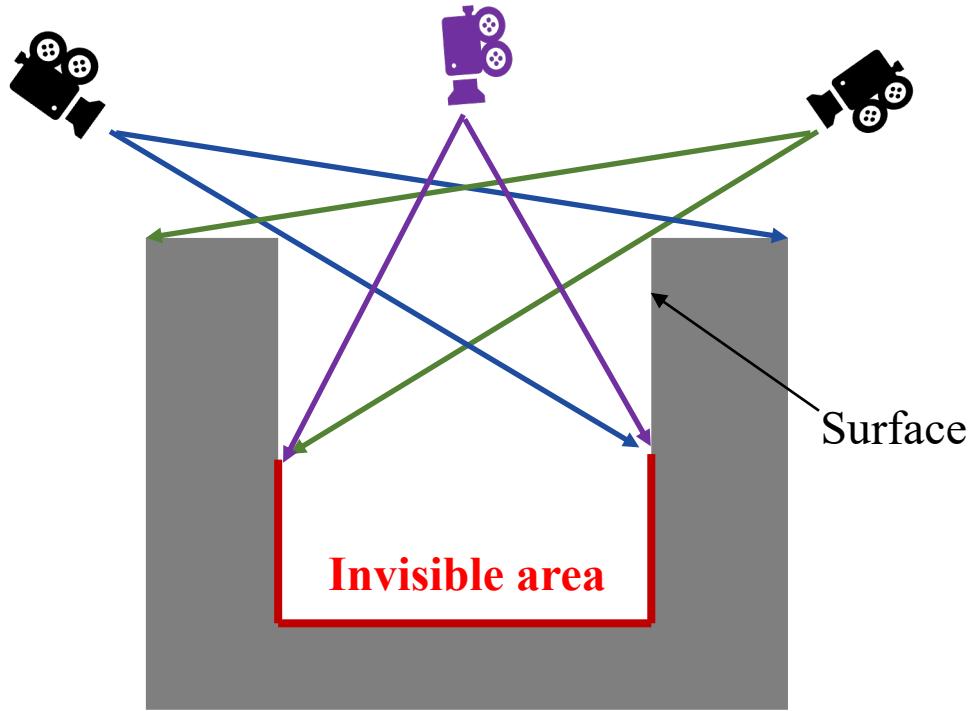
Note: Here , we refer to the inria's version of 3DGS;
NOT those improved models such as sparse-view 3DGS, fast-training 3DGS, 3DGS compression, *etc.*

Pipeline of Feed-Forward 3DGS

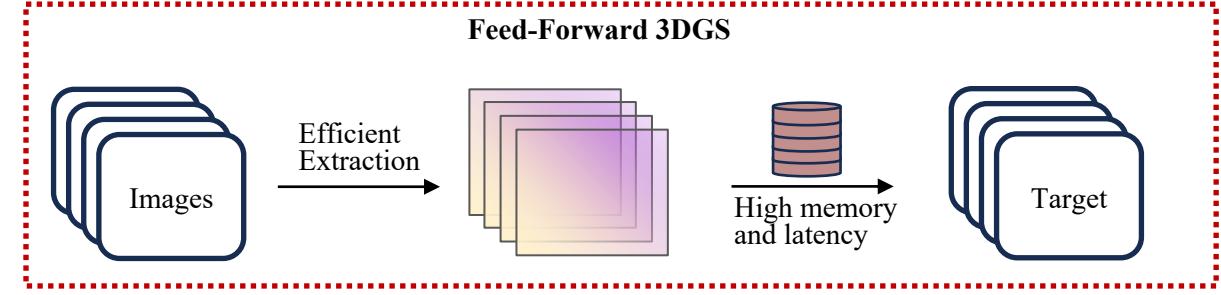


Almost all feed-forward 3DGS networks use this paradigm.

Challenges in Feed-Forward 3DGS

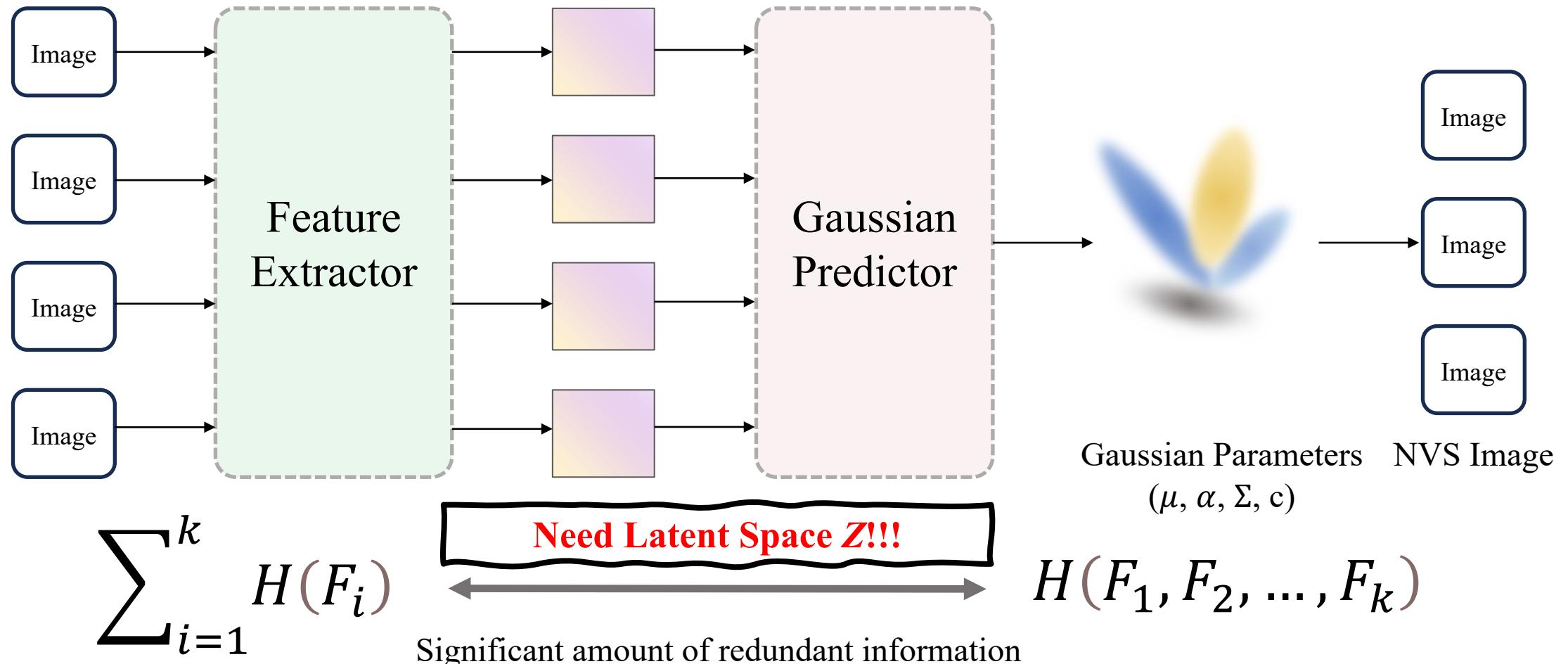


We need denser views to **provide more information**, but at the same time not be influenced by **redundancy**.

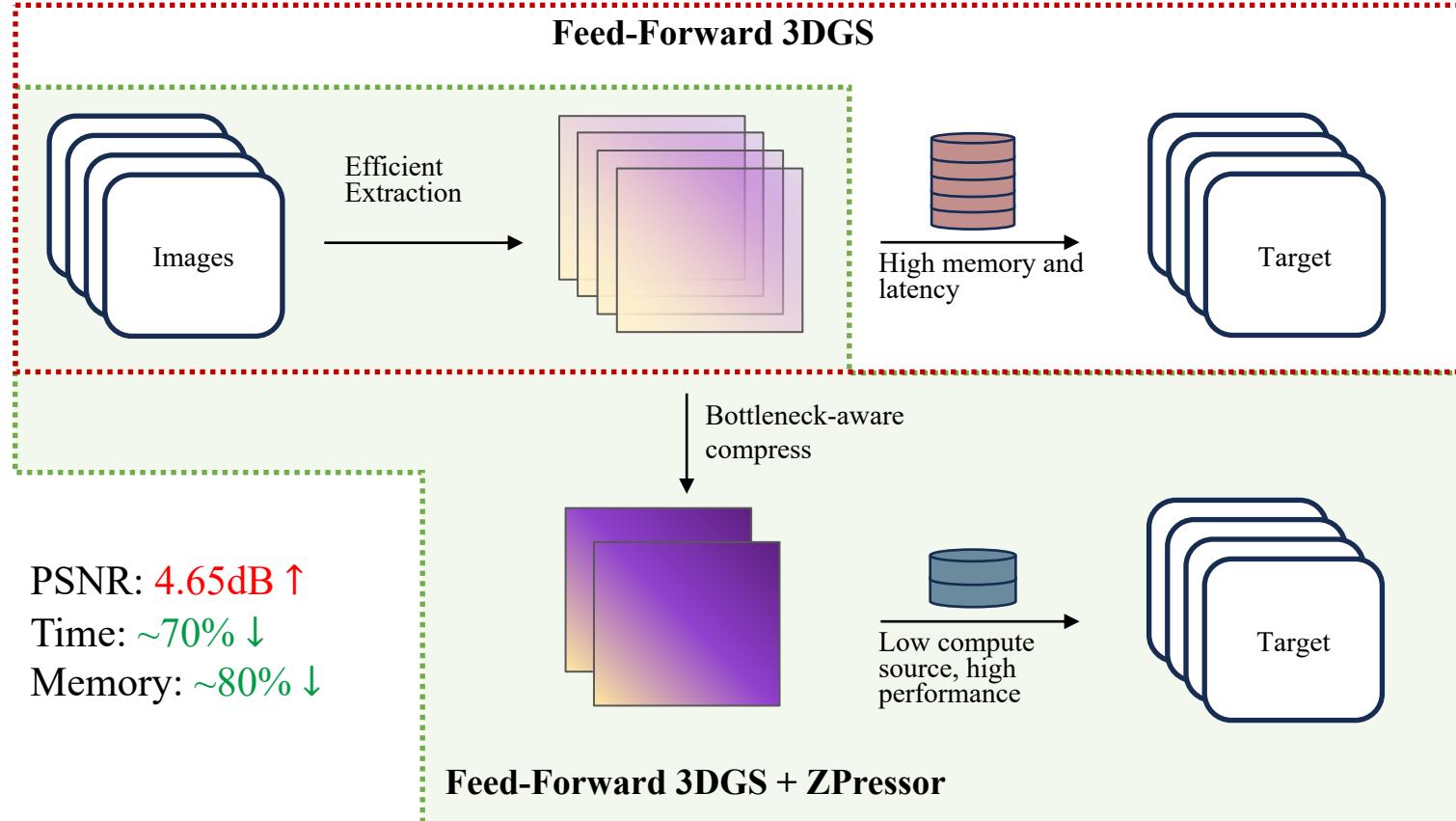


The scalability of feed-forward 3DGS is fundamentally constrained by the **limited capacity** of their networks.

Information Flow in FF 3DGS



Bottleneck-Aware Compression

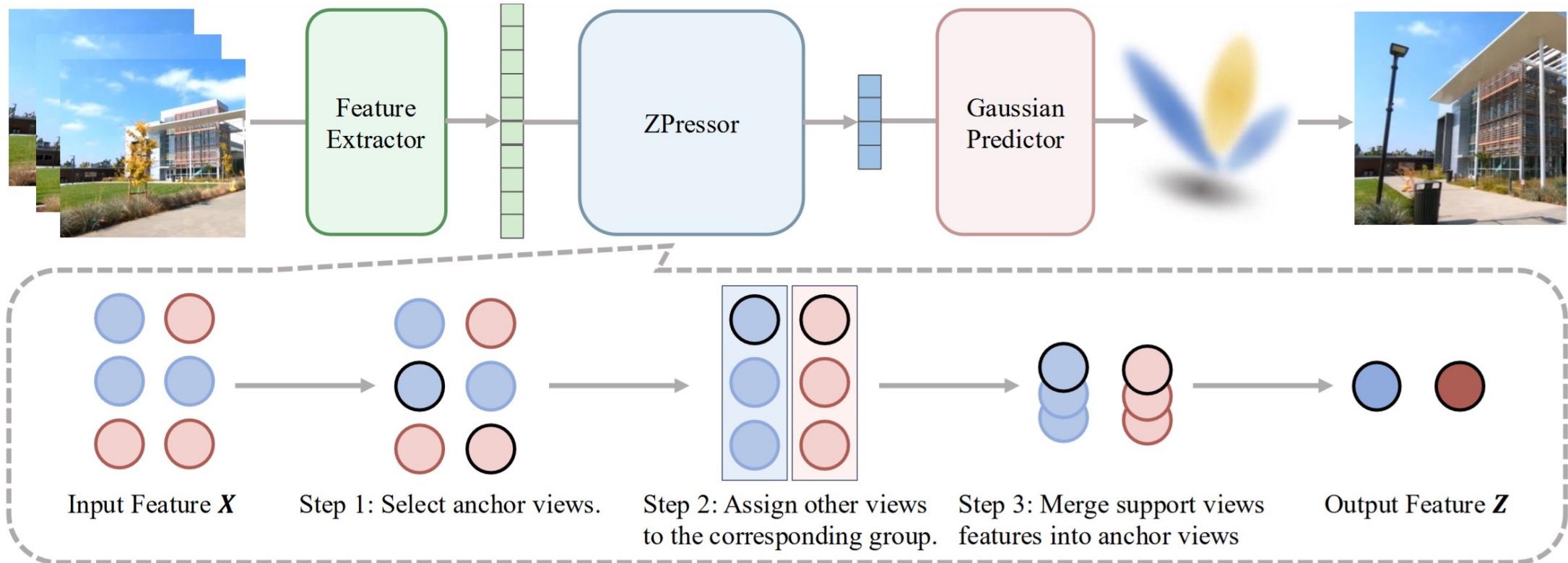


$$\min_{\mathcal{Z}} IB = \underbrace{\beta I(\mathcal{X}, \mathcal{Z})}_{\text{Compression Score}} - \underbrace{I(\mathcal{Z}, \mathcal{Y})}_{\text{Prediction Score}}$$

- 1. Compression Score:** Minimizing $I(\mathcal{X}, \mathcal{Z})$
- 2. Prediction Score:** Maximizing $I(\mathcal{Z}, \mathcal{Y})$

Note: The mutual information (MI) of two random variables $I(\cdot, \cdot)$ is a measure of the mutual dependence between the two variables.

Zpressor: Overview



Anchor View Selection

Support-to-anchor Assignment

Views Information Fusion

Anchor View Selection

Algorithm 2 Farthest Point Sampling for Anchor View Selection

Input: Set of view camera positions $\mathcal{T} = \{\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_K\}$, Number of anchor views N

Output: Indices of the selected anchor views $\mathcal{S} = \{\mathbf{T}_{a_1}, \mathbf{T}_{a_2}, \dots, \mathbf{T}_{a_n}\}$

Initialize the set of anchor view indices $\mathcal{S} \leftarrow \emptyset$

Randomly select a random anchor view $\mathbf{T}_{a_1} \in \mathcal{T}$, where $\mathbf{T}_{a_1} \sim \text{Uniform}(\mathcal{T})$

Add \mathbf{T}_{a_1} to \mathcal{S} : $\mathcal{S} \leftarrow \{\mathbf{T}_{a_1}\}$

for $j \leftarrow 2$ to N **do**

 Initialize a dictionary to store minimum distances $D \leftarrow \{\}$

for $k \leftarrow 1$ to K **do**

if $k \notin \mathcal{S}$ **then**

 Calculate the minimum distance $d_k \leftarrow \min_{i \in \mathcal{S}} \|\mathbf{T}_k - \mathbf{T}_i\|_2$

 Store the distance: $D[k] \leftarrow d_k$

end if

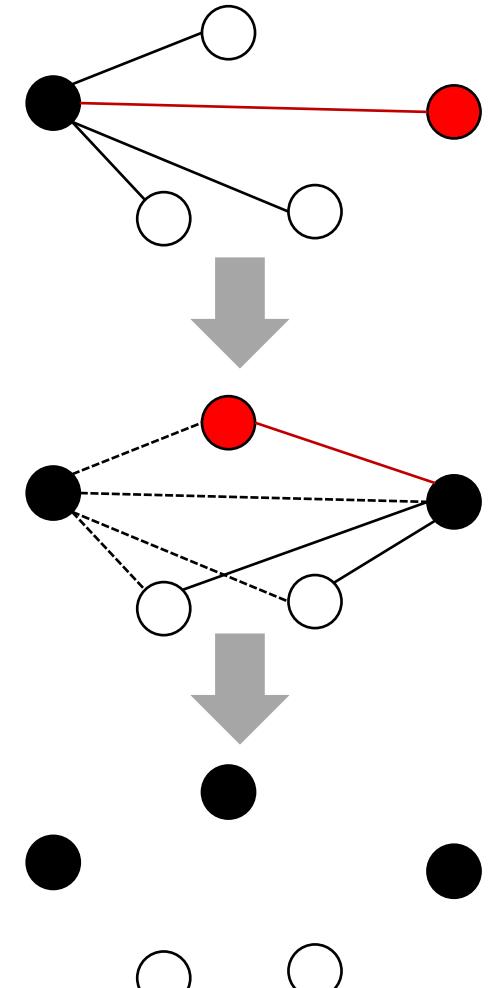
end for

 Find the view position \mathbf{T}_{a_j} with the maximum minimum distance: $\mathbf{T}_{a_j} \leftarrow \arg \max_{k \notin \mathcal{S}} D[k]$

 Add a_j to \mathcal{S} : $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathbf{T}_{a_j}\}$

end for

return \mathcal{S}



K=5; N=3

Support-to-anchor Assignment

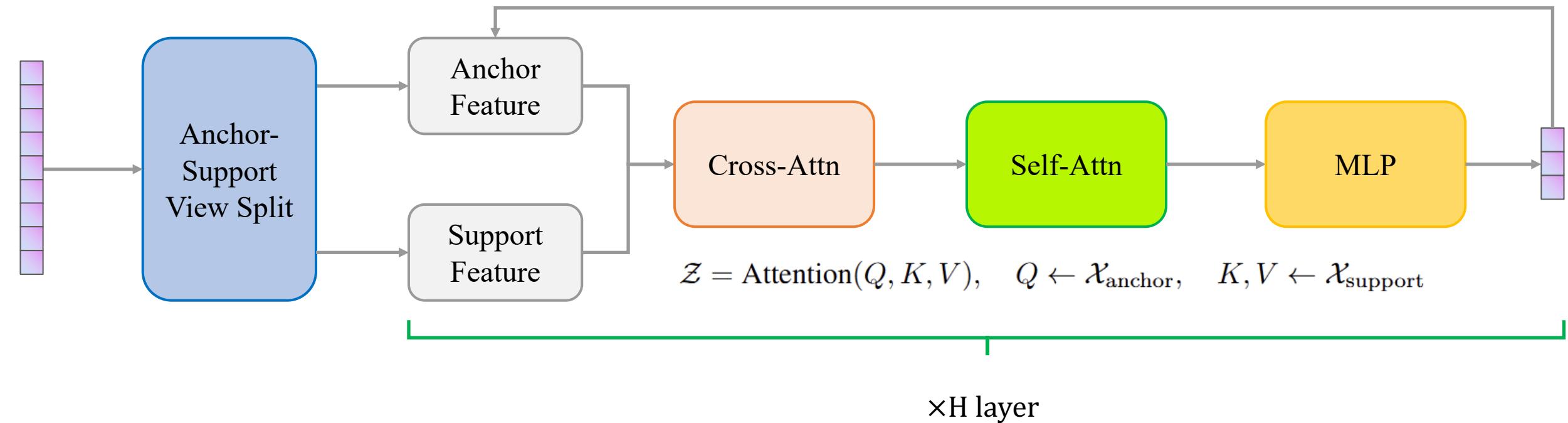


View Groups after Step 1 and Step 2

- Once anchor views are selected, each support view is assigned to its nearest anchor based on **camera position**.
- This grouping ensures that support views, which capture complementary scene details, are paired with **the most spatially relevant** anchor views.
- This pairing thereby ensures the effectiveness of information fusion.
- Formally, the cluster assignment to the i -th anchor view can be denoted as:

$$\mathcal{C}_i = \{f(\mathbf{T}) \in \mathcal{X}_{\text{support}} \mid \|\mathbf{T} - \mathbf{T}_{a_i}\| \leq \|\mathbf{T} - \mathbf{T}_{a_j}\|, \forall j \neq i\}$$

Views Information Fusion



Design of Feature Fusion Networks. Feature Fusion by Cross-Attention,
Self-Attention and MLP.

Results on DL3DV with DepthSplat

Views	Methods	PSNR↑	SSIM↑	LPIPS↓
36 views	DepthSplat	19.23	0.666	0.286
	DepthSplat + ZPressor	23.88 _{+4.65}	0.815 _{+0.149}	0.150 _{-0.136}
24 views	DepthSplat	20.38	0.711	0.253
	DepthSplat + ZPressor	24.26 _{+3.88}	0.820 _{+0.109}	0.147 _{-0.106}
16 views	DepthSplat	22.07	0.773	0.195
	DepthSplat + ZPressor	24.25 _{+2.18}	0.819 _{+0.046}	0.147 _{-0.047}
12 views	DepthSplat	23.32	0.807	0.162
	DepthSplat + ZPressor	24.30 _{+0.97}	0.821 _{+0.014}	0.146 _{-0.017}

Results on RE10K with MVsplat and pixelSplat

Views	Methods	PSNR↑	SSIM↑	LPIPS↓
36 views	pixelSplat	OOM	OOM	OOM
	pixelSplat + ZPressor	26.59	0.849	0.225
	MVsplat	24.19	0.851	0.155
	MVsplat + ZPressor	27.34_{+3.15}	0.893_{+0.042}	0.113_{-0.042}
24 views	pixelSplat	OOM	OOM	OOM
	pixelSplat + ZPressor	26.72	0.851	0.223
	MVsplat	25.00	0.871	0.137
	MVsplat + ZPressor	27.49_{+2.49}	0.895_{+0.024}	0.111_{-0.026}
16 views	pixelSplat	OOM	OOM	OOM
	pixelSplat + ZPressor	26.81	0.853	0.221
	MVsplat	25.86	0.888	0.120
	MVsplat + ZPressor	27.60_{+1.74}	0.896_{+0.008}	0.110_{-0.010}
8 views	pixelSplat	26.19	0.852	0.215
	pixelSplat + ZPressor	26.86_{+0.67}	0.854_{+0.002}	0.219_{+0.004}
	MVsplat	26.94	0.902	0.107
	MVsplat + ZPressor	27.72_{+0.78}	0.897_{-0.005}	0.109_{+0.002}

Qualitative comparison



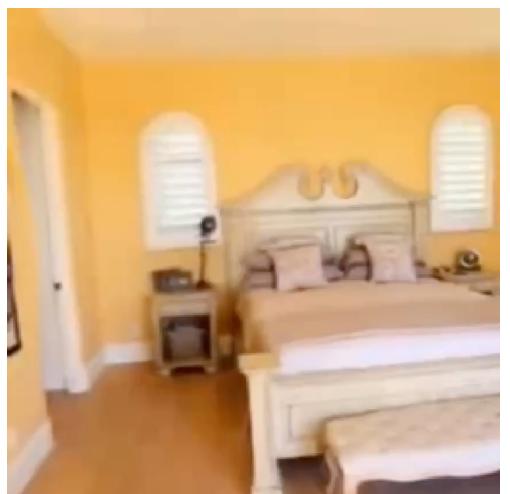
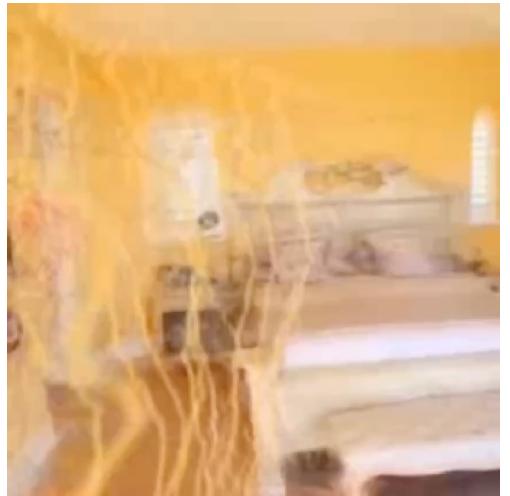
DepthSplat

DepthSplat+ZPressor

DepthSplat

DepthSplat+ZPressor

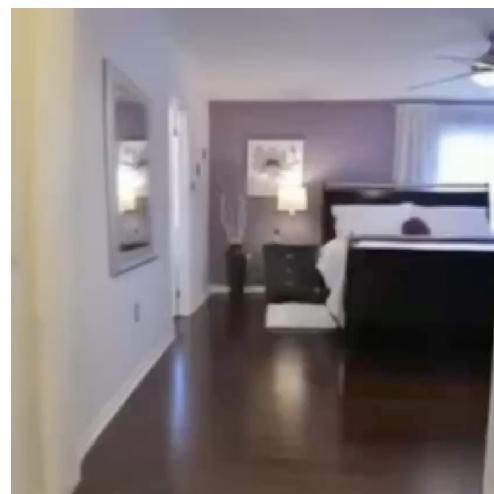
Qualitative comparison



MVSplat



MVSplat+ZPressor



MVSplat

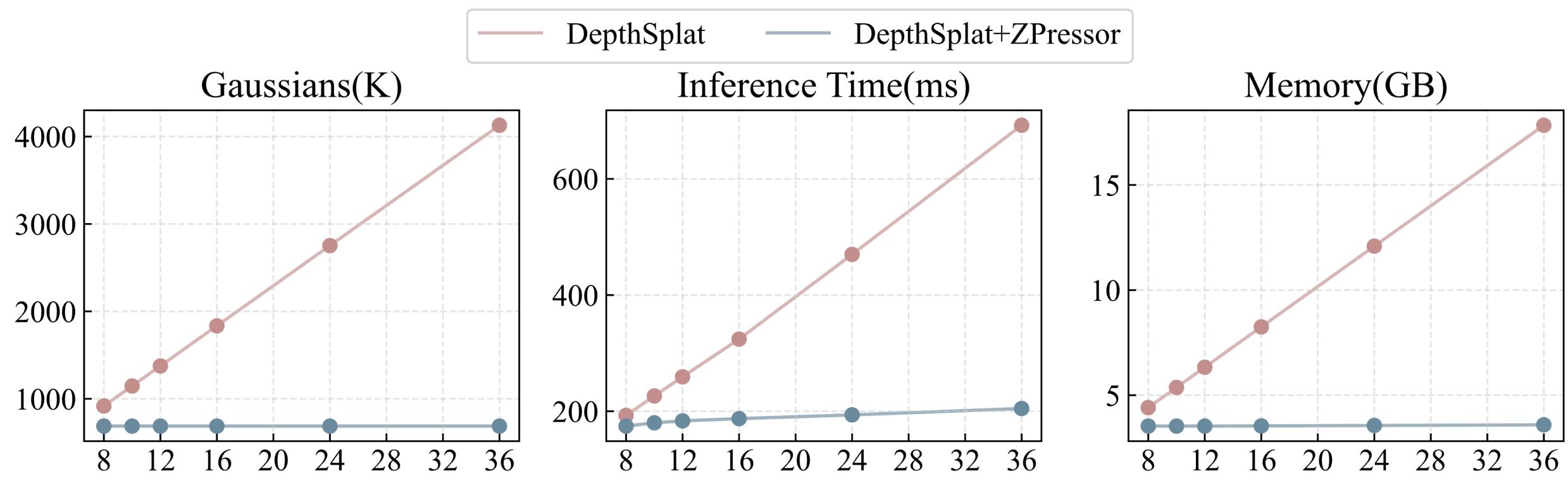


MVSplat+ZPressor

Cross Dataset Generalization on ACID

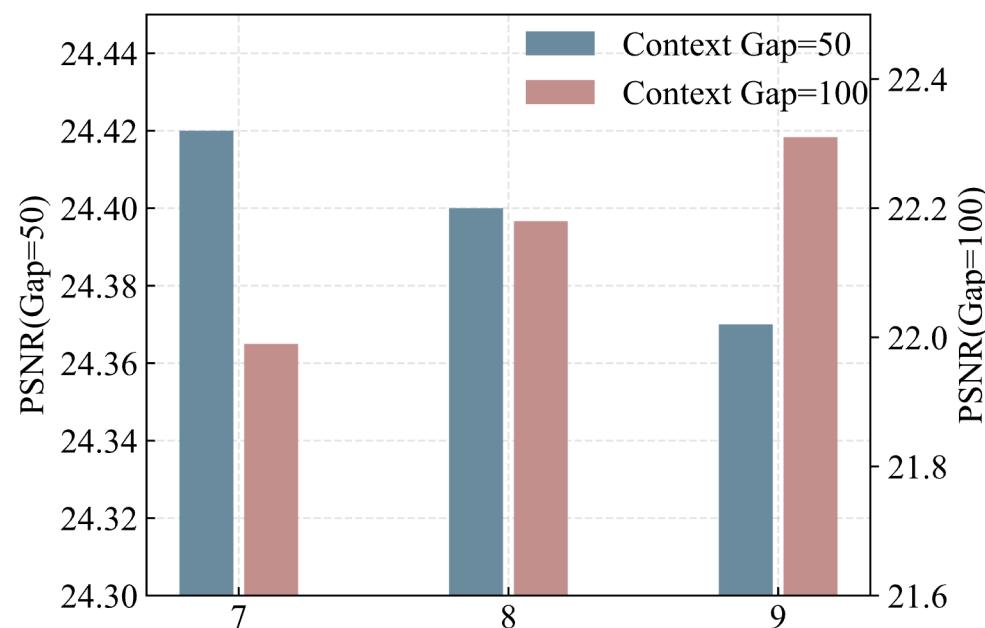
Views	Methods	PSNR↑	SSIM↑	LPIPS↓
36 views	pixelSplat	OOM	OOM	OOM
	pixelSplat + Ours	27.78	0.823	0.238
	MVSplat	24.89	0.812	0.179
	MVSplat + Ours	28.16 _{+3.27}	0.853 _{+0.041}	0.145 _{-0.034}
24 views	pixelSplat	OOM	OOM	OOM
	pixelSplat + Ours	27.91	0.825	0.235
	MVSplat	25.46	0.829	0.167
	MVSplat + Ours	28.33 _{+2.87}	0.856 _{+0.027}	0.142 _{-0.025}
16 views	pixelSplat	OOM	OOM	OOM
	pixelSplat + Ours	27.97	0.826	0.234
	MVSplat	26.08	0.844	0.156
	MVSplat + Ours	28.42 _{+2.34}	0.858 _{+0.014}	0.141 _{-0.015}
8 views	pixelSplat	26.69	0.807	0.260
	pixelSplat + Ours	28.05 _{+1.36}	0.828 _{+0.021}	0.234 _{-0.026}
	MVSplat	27.89	0.864	0.140
	MVSplat + Ours	28.60 _{+0.71}	0.860 _{-0.004}	0.140 _{-0.000}

Model Efficiency



Linear no more: constant memory, constant time.

Bottleneck Analysis and Ablation Study



Analysis of bottleneck:

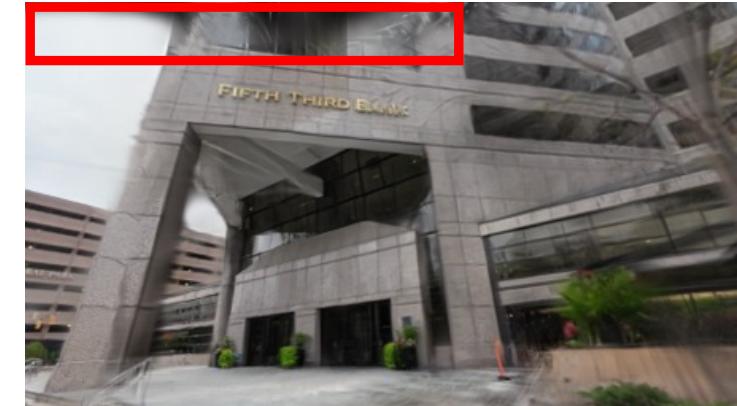
- Different levels of complexity benefit from different bottlenecks
- Effective compression preserves essential scene information.

Methods	PSNR↑	SSIM↑	LPIPS↓	Time (s)	Peak Memory (GB)
DepthSplat + ZPressor	24.30	0.821	0.146	0.184	3.80
w/o multi-blocks	24.18	0.817	0.149	0.140	3.79
w/o self-attention	23.85	0.810	0.156	0.183	3.80
DepthSplat	23.32	0.808	0.162	0.260	6.80

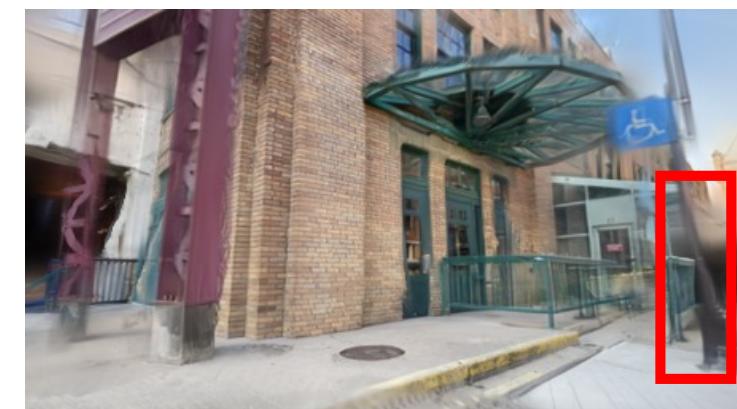
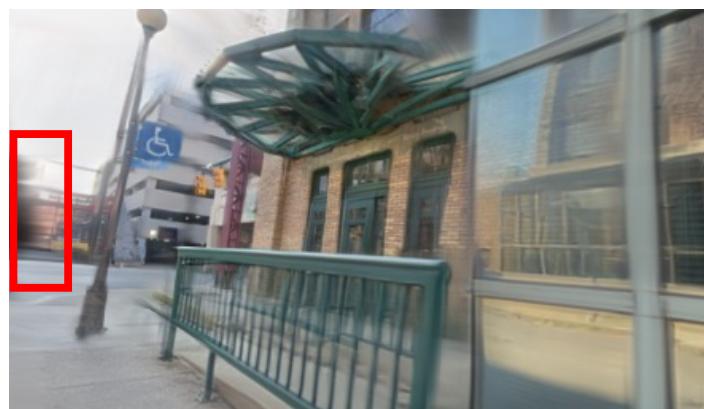
Limitations



⋮



⋮



Inputs (~500 views)

DepthSplat + ZPressor

ZPressor exhibits limitations when processing scenarios with an **extremely high** density of input views.

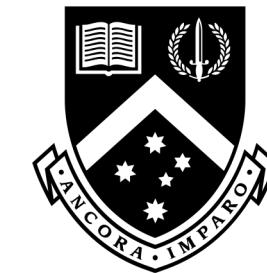
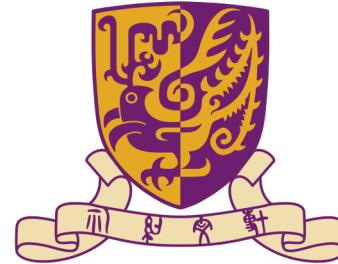
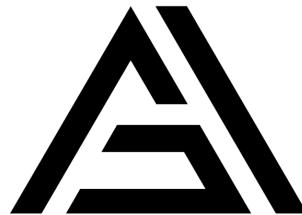
VolSplat

Weijie Wang^{1,2*} Yeqing Chen^{3*} Zeyu Zhang² Hengyu Liu^{2,4} Haoxiao Wang¹ Zhiyuan Feng⁵
Wenkang Qin² Zheng Zhu^{2†} Donny Y. Chen⁶ Bohan Zhuang^{1†}

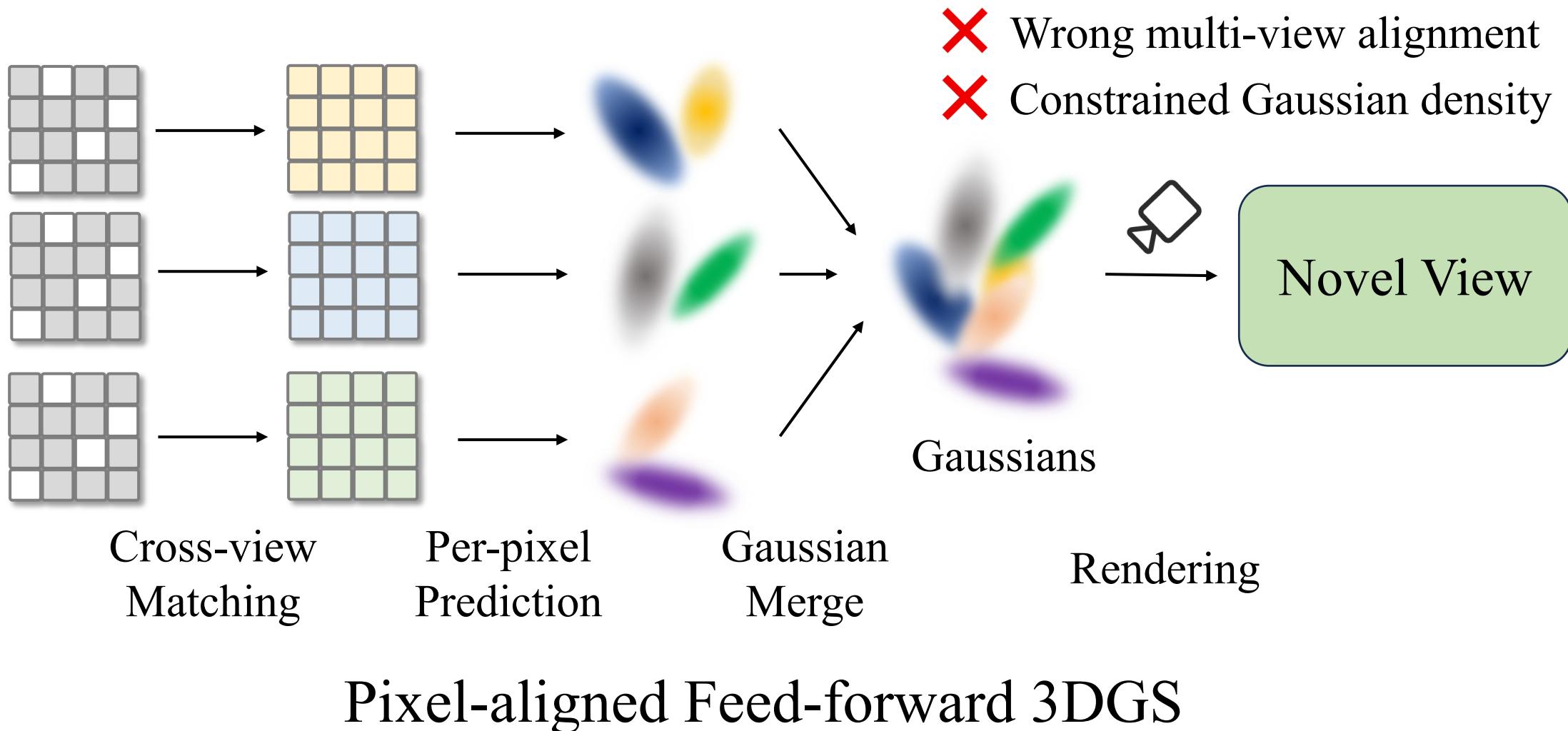
* Equal contribution † Corresponding authors

¹Zhejiang University ²GigaAI ³University of Electronic Science and Technology of China

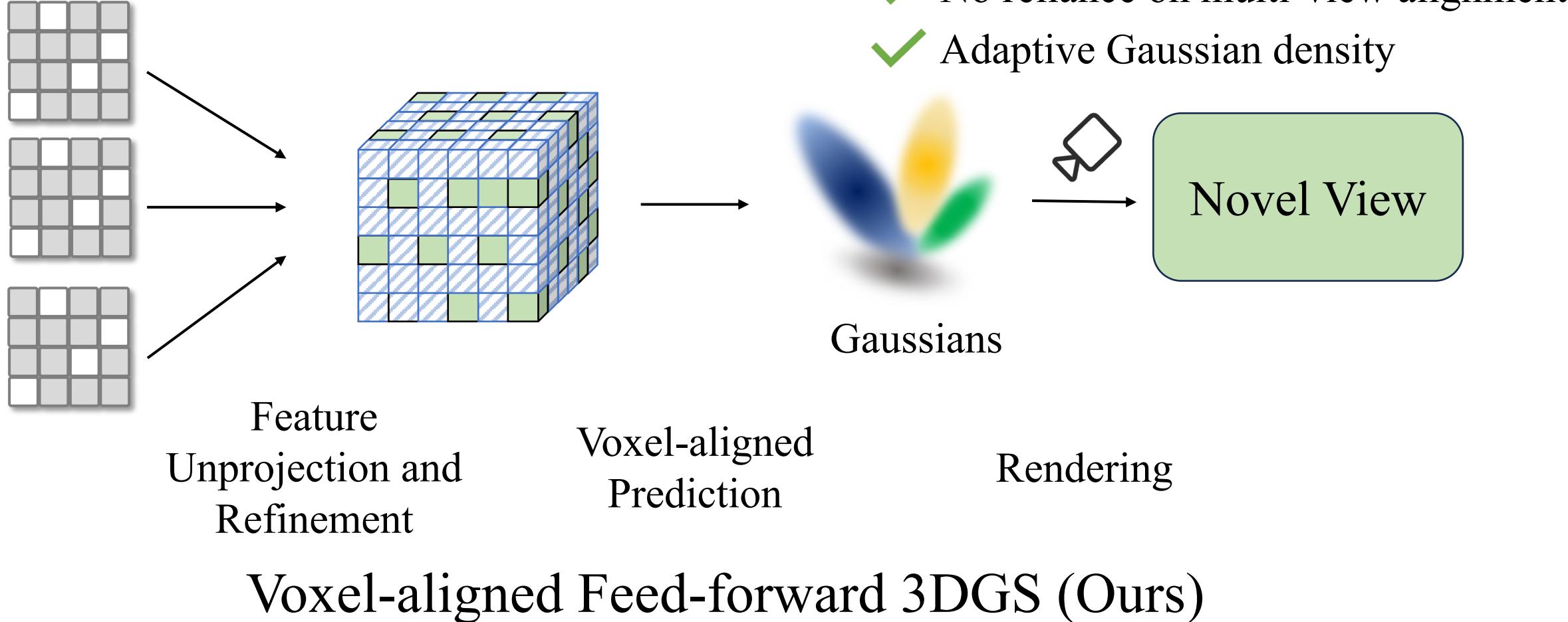
⁴The Chinese University of Hong Kong ⁵Tsinghua University ⁶Monash University



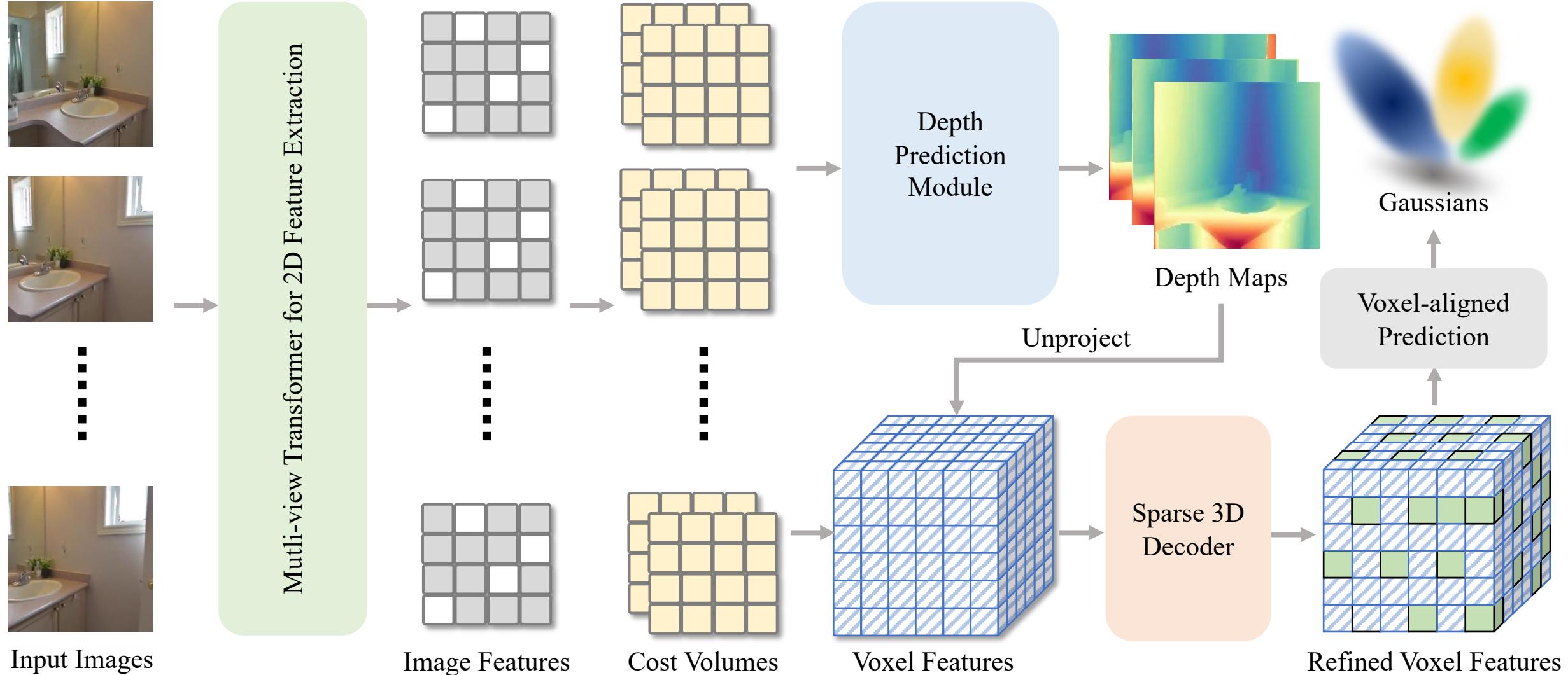
Previous Feed-Forward Methods



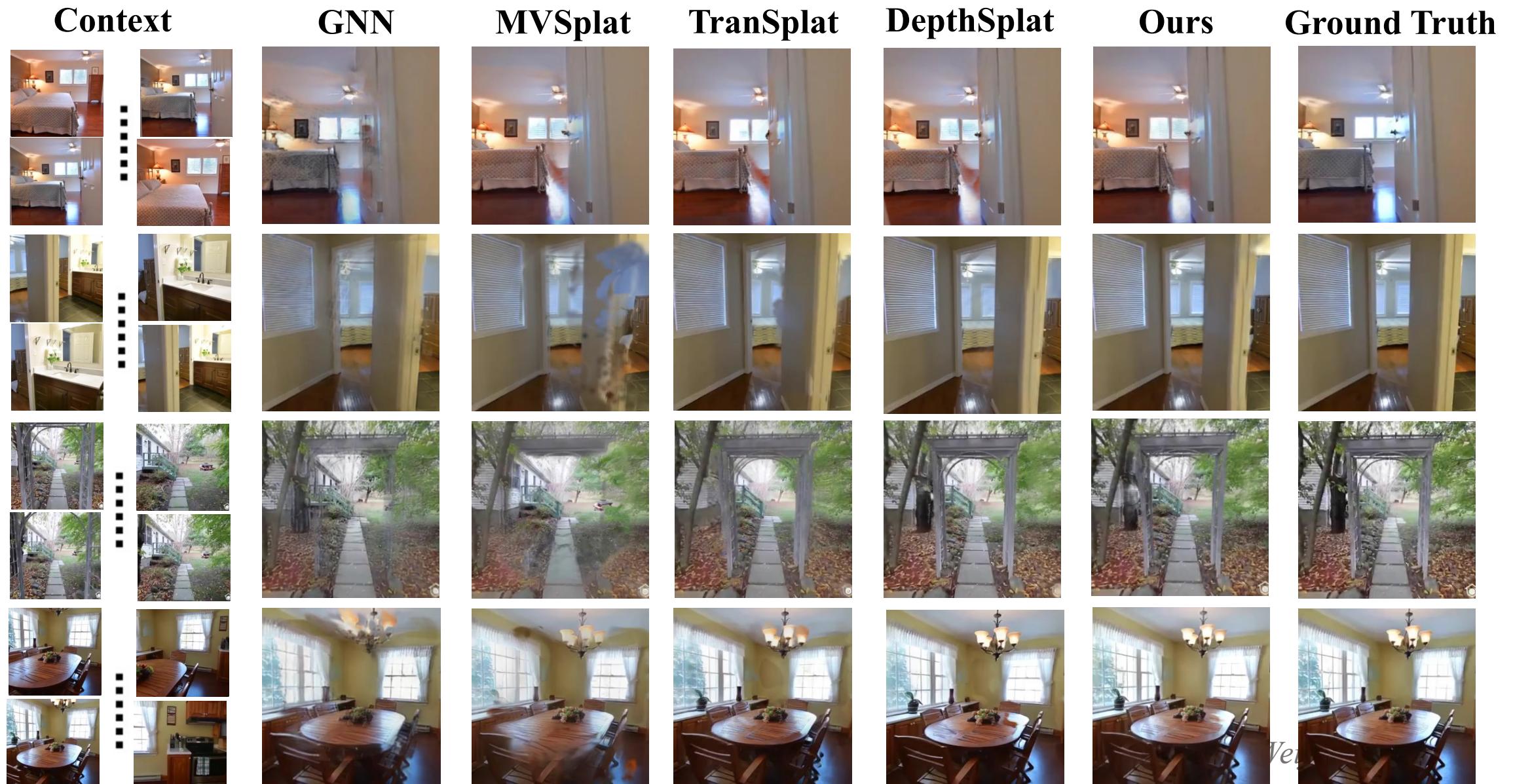
volSplat



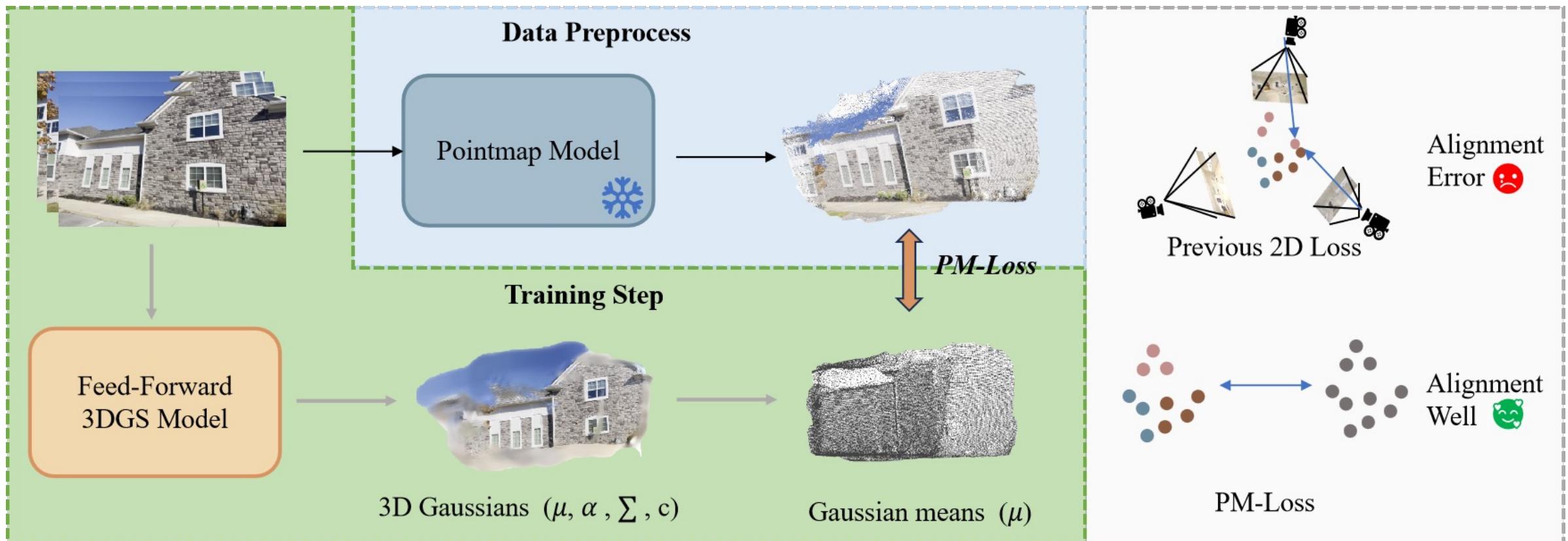
Pipeline



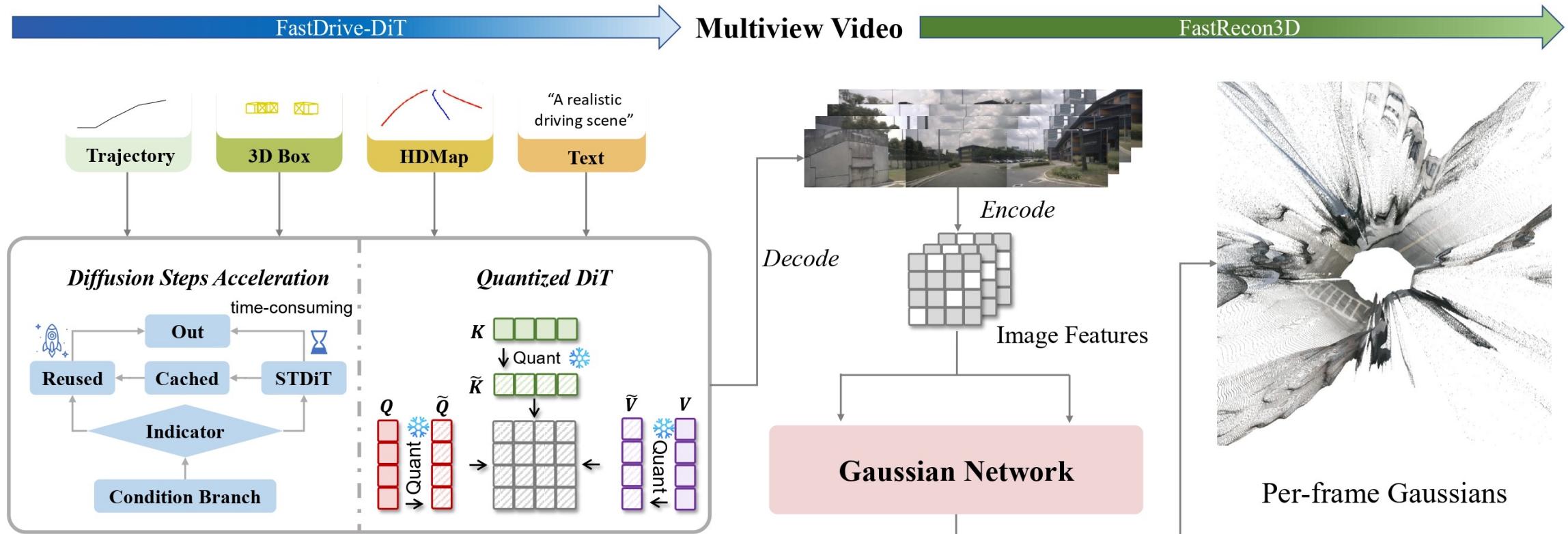
SoTA Performance



PM-LOSS



DriveGen3D



More Information



ZPressor's project page.
Paper, code and models
are available.



Weijie Wang's WeChat.
Actively seeking
internship opportunities.

Conclusion:

- ZPressor is a **lightweight, architecture-agnostic** module designed for scalable feed-forward 3DGS
- We bridges IB principle and 3D generative modeling, offering a new perspective on scalable 3D scene reconstruction.