GaussianLens: Localized High-Resolution Reconstruction via On-Demand Gaussian Densification

Yijia Weng[1⋆] Zhicheng Wang[2] Songyou Peng[2]
Saining Xie[2] Howard Zhou[2] Leonidas Guibas[2,1]

[1]Stanford University [2]Google Deepmind
https://yijiaweng.github.io/gaussianlens/

Abstract. We perceive our surrounding environments with an active focus, paying more attention to regions of interest, such as the shelf labels in a grocery store or a family photo on the wall. When it comes to scene reconstruction, this human perception trait calls for spatially varying degrees of detail ready for closer inspection in critical regions, preferably reconstructed on demand as users shift their focus. While recent approaches in 3D Gaussian Splatting (3DGS) can achieve fast, generalizable scene reconstruction from sparse views, their uniform resolution output leads to high computational costs, making them unscalable to high-resolution training. As a result, they cannot leverage available image captures at their original high resolution for detail reconstruction. Per-scene optimization methods reconstruct finer details with heuristic-based adaptive density control, yet require dense observations and lengthy offline optimization. To bridge the gap between the prohibitive cost of high-resolution holistic reconstructions and the user needs for localized fine details, we propose the problem of localized high-resolution reconstruction through on-demand generalizable Gaussian densification. Given an initial low-resolution 3DGS reconstruction, the goal is to learn a generalizable network that densifies the reconstruction to capture fine details in a user-specified local region of interest (RoI), based on sparse high-resolution observations of the RoI. This formulation avoids the high cost and redundancy of uniformly high-resolution reconstructions and enables the full leverage of high-resolution observations in critical regions. To address the problem, we propose *GaussianLens*, a feed-forward densification framework that fuses multi-modal information from the initial 3DGS and multi-view images. We further propose a pixel-guided densification mechanism that effectively captures details under significant resolution increases. Experiments demonstrate our method's superior performance in local high-fidelity detail reconstruction and strong scalability to images of up to 1024×1024 resolution.

Keywords: 3D Gaussian Splatting · Neural Reconstruction

1 Introduction

3D Gaussian Splatting (3DGS) [15, 17, 55] has shown great promise in photorealistic 3D reconstruction and novel-view synthesis. More recently, general-

⋆ Work began during an internship at Google DeepMind

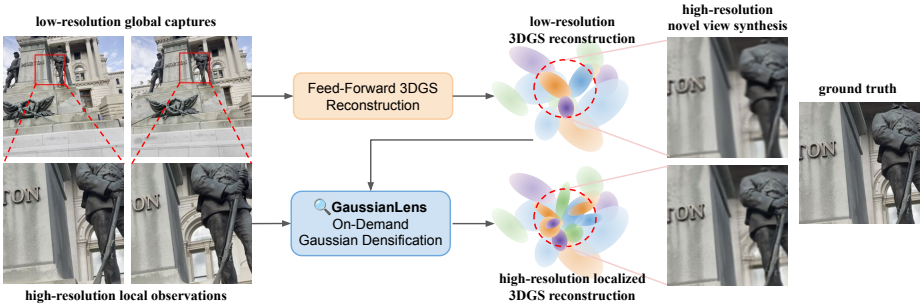

Fig. 1: We introduce the problem of localized high-resolution reconstruction via on-demand Gaussian densification. While the majority of feed-forward models are confined to single-pass, uniform-resolution reconstruction, *GaussianLens* achieves low-cost, high-resolution local reconstruction by learning to densify low-resolution initial 3DGS reconstructions conditioned on high-resolution local observations.

izable 3DGS reconstruction methods [2, 3, 35–37, 42, 49, 57, 59] have extended these capabilities to sparse input views and on-the-fly reconstruction settings. Most existing works predict Gaussians with a regular structure, typically pixel-aligned [2, 4, 36, 37, 47, 59] or voxel-aligned [3, 58]. While structured outputs facilitate learning, they lead to a uniform-resolution reconstruction, with the same number of Gaussians allocated to each pixel or voxel. This approach is inefficient, as capturing fine details requires a computationally costly global increase in the reconstruction resolution, making it impractical to leverage readily available high-resolution observations. For example, while DL3DV [23] contains 3840×2160 videos, existing novel-view synthesis works only use up to 960×540, with many defaulting to 256×256. Training state-of-the-art DepthSplat [47] on 1024×1024 inputs is also infeasible as it cannot fit into the memory of an 80GB H100 GPU. In contrast, per-scene optimization methods produce non-uniform reconstructions adapted to varying scene complexity via Adaptive Density Control [17], a heuristic mechanism to densify Gaussians in under-reconstructed regions. Yet they rely on dense observations and lengthy offline optimization. To sum up, existing methods are bottlenecked by computational costs to fully leverage all available high-resolution images for reconstruction, especially in fast, interactive settings.

Meanwhile, uniformly high-resolution reconstruction of the entire scene is often unnecessary. Humans typically focus on the fine details within a small region of interest, while a lower-resolution view of the surroundings suffices for a holistic understanding. For example, in an interactive room capture, the user may want to ensure the titles on book spines are legible, but care less about the tiny dents on the floor. This perceptual trait calls for a more efficient approach: reconstructions with spatially varying degrees of detail, where critical areas are reconstructed at higher fidelity. To reduce redundancy and cost, these details are ideally reconstructed on demand as the user's focus shifts.

To bridge the gap between prohibitively costly high-resolution holistic reconstructions and the user needs for localized fine details, we introduce the problem of localized high-resolution reconstruction through on-demand Gaussian densification. Given a low-resolution 3DGS reconstruction of the entire scene, the goal is to densify a user-specified region of interest to reveal finer details, guided by a sparse set of high-resolution images of the region. The region is specified by 2D masks, mimicking the scenario where a user selects an area from the current view to "zoom in" for more details. The reconstruction is evaluated on high-resolution novel view renderings of the selected region.

To address the problem, we introduce *GaussianLens*, a cross-modal framework that aggregates information from 2D images to the initial 3D Gaussians via complementary mechanisms. We first render the initial 3D Gaussians and compare them to groundtruth observations to obtain residual information, based on which we construct initial Gaussian features and image features. We then use a PointTransformerV3 [43]-based encoder for Gaussian feature extraction, where we employ projection-based image-Gaussian cross-attention layers to further fuse information from images. Finally, we decode Gaussian features into densified Gaussian parameters, expressed as offsets from initial Gaussian parameters. The whole process can be viewed as a learned version of the densify-by-clone step and subsequent optimizations in per-scene 3DGS reconstruction.

However, with larger resolution increases, cloning-based densification struggles to capture all newly introduced details. We therefore propose *pixel-guided densification*, where we spawn one Gaussian for each high-resolution pixel in the region of interest. These Gaussians effectively preserve high-resolution details and complement the input Gaussians that serve as a coarse scaffold.

To summarize, our main contributions are:

- We propose the problem of localized high-resolution reconstruction via on-demand Gaussian densification, a formulation that avoids the prohibitive and unnecessary cost of uniformly high-resolution reconstructions and enables full leverage of high-resolution local observations.
- We develop *GaussianLens*, a multi-modal framework featuring complementary mechanisms to fuse information from 3D Gaussians and multi-view images for effective densification prediction.
- We propose a novel pixel-based densification mechanism that better captures details under substantial resolution increase in the one-step feed-forward densification scenario.

We build a benchmark for the proposed problem based on RealEstate10K [63] and DL3DV [23], and compare our method against state-of-the-art generalizable 3DGS reconstruction methods. We achieve efficient on-demand high-resolution detail reconstruction in local regions with qualities above or on par with uniformly high-resolution models, while using fewer computational resources. We can also leverage up to 1024×1024 high-resolution observations, which uniform feed-forward models cannot scale up to.

2 Related Work

2.1 Generalizable 3D Gaussian Prediction

Generalizable 3D Gaussian prediction methods learn to predict 3D Gaussian reconstructions with a network forward pass, typically conditioned on sparse observations. Significant progress has been made to reconstruct objects [3, 26, 32,36,37,49,58,59,64] and scenes [2,4,24,35,42,47,57] under sparse-view settings. Integration with large 2D [1] or geometric [21,40] foundation models enables 360° view synthesis [5], semantic [11] and pose-free reconstructions [10, 14, 16, 22, 51]. Most methods predict a structured set of Gaussians, typically pixel-aligned [2,4, 36,37,47,59] or voxel-aligned [3,58]. While structured outputs facilitate learning, their uniform resolution limits their ability to reconstruct high-resolution details due to the high computational cost. An exception, PanSplat [56], specifically targets 4K (2048×4096) panorama synthesis. It supports up to 768×1536 with efficient hierarchical cost volume and Gaussian head designs, but scaling to 4K is only achieved with deferred backpropagation.

2.2 Generalizable 3D Gaussian Update

More recently, generalizable frameworks have been used to update given initial Gaussians. [6] learns to iteratively update Gaussians by leveraging cues from rendering gradients. [7] trains a point transformer to refine flawed 3D Gaussians to reduce artifacts at out-of-distribution views. Most related to us is Generative Densification (GD) [28], which proposes to attach a learned Gaussian densification module to feed-forward frameworks to improve the reconstruction in high-frequency, under-reconstructed regions. However, GD consumes latent features from the base feed-forward model, and has to be tailored to and fine-tuned with it. The resulting model still performs single-pass image-to-Gaussian prediction for the full scene. In contrast, our model is source-agnostic, operating directly on the Gaussians with no access to or assumptions about the source model. We also have the flexibility to build on an existing reconstruction and densify exclusively in specified local regions.

2.3 Adaptive Density Control

In per-scene 3D Gaussian optimization, Adaptive Density Control [17] enables efficient reconstruction of spatially varying scene details, by heuristically selecting and densifying Gaussians in under-reconstructed regions. Many methods have been proposed to improve the selection heuristic [8, 19, 27, 30, 52, 62] and the initialization of new Gaussians [8,19,27,30]. However, they all require dense observations and time-consuming per-scene optimization.

2.4 Multi-Scale Gaussian Splatting

Multi-scale and hierarchical Gaussian Splatting methods reconstruct the scene with layers of Gaussians capturing scene details at different scales. During rendering, levels of details are flexibly chosen based on computational resources

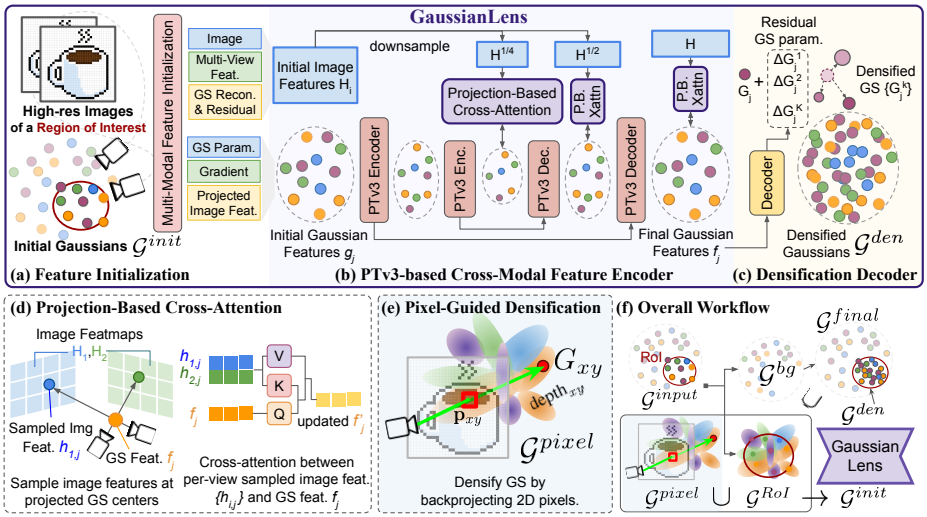

Fig. 2: Method overview. (a)-(d) illustrates *GaussianLens*, our feed-forward densification framework. It constructs multi-modal features for initial Gaussians and images (a), further extracts features via a PTv3-based encoder (b) with projection-based cross attention (d) to images, and decodes them into residual parameters of densified Gaussians (c). (e) presents pixel-guided densification. (f) shows the overall workflow.

and user needs, allowing large-scale scene reconstruction [18, 25, 29], anti-aliased view-adaptive rendering [9, 31, 34, 50], and generalizable coarse-to-fine scene reconstruction [38]. They require lengthy offline optimization and costly storage of the full Gaussian hierarchy, before *rendering* with flexible level-of-detail. In contrast, we aim to support on-demand level-of-detail at the *reconstruction* stage.

2.5 Super-Resolution and Close-Up Novel View Synthesis

Most super-resolution and close-up novel view synthesis methods reconstruct high-resolution details by distilling 2D image super-resolution models [12, 20, 33, 39, 44–46, 53, 54]. While they all rely on slow per-scene optimization, we reconstruct fine details with a fast network forward pass. Closer to our setting is [61], which proposes a unified frequency-aware radiance field to capture both normal-resolution global scene structure and tiny details in an area of interest, given high-resolution captures of the area. We share their goal of leveraging local high-resolution observations, but pursue it with generalizable 3D Gaussians.

3 Method

Given a set of 3D Gaussians $\mathcal{G}^{input}$ reconstructed from low-resolution input images, our goal is to densify and refine $\mathcal{G}^{input}$ to reconstruct high-resolution

details in a user-specified local region of interest (RoI) based on additional high-resolution images of the specified region.

Concretely, we start with a sparse set of low-resolution input images $\mathcal{I}^{low}$ with known cameras, and apply state-of-the-art feed-forward 3DGS reconstruction method DepthSplat [47] to obtain the initial 3DGS reconstruction $\mathcal{G}^{input}$.

We take another sparse set of high-resolution images capturing the RoI, denoted by $\mathcal{I} = \{I_i\}_{i=1}^N, I_i \in \mathbb{R}^{H\times W\times 3}$, with known camera projection matrices $\{\mathbf{P}_i\}_{i=1}^N$. The RoI is specified by its 2D projections onto the input views, given as binary masks $\mathcal{M} = \{M_i\}_{i=1}^N, M_i \in \{0,1\}^{H\times W}$.

To tackle the problem, we design a cross-modal Gaussian densification prediction framework *GaussianLens* (Sec. 3.1). It fuses multi-modal information from a set of initial Gaussians $\mathcal{G}^{init}$ and 2D images, and predicts a set of densified Gaussians $\mathcal{G}^{den}$. We divide the input Gaussians $\mathcal{G}^{input}$ into those in the RoI ($\mathcal{G}^{RoI}$) and the background ($\mathcal{G}^{bg}$), apply *GaussianLens* to $\mathcal{G}^{RoI}$ for local densification, i.e. $\mathcal{G}^{init} \leftarrow \mathcal{G}^{RoI}$, and merge the results with the background, i.e., $\mathcal{G}^{final} = \mathcal{G}^{den} \cup \mathcal{G}^{bg}$.

To better reconstruct details, we propose a novel pixel-guided Gaussian densification mechanism (Sec. 3.2) that augments $\mathcal{G}^{RoI}$ with another set of pixel-guided Gaussians $\mathcal{G}^{pixel}$, before applying *GaussianLens* to their union, i.e., $\mathcal{G}^{init} \leftarrow \mathcal{G}^{RoI} \cup \mathcal{G}^{pixel}$. The full workflow is shown in Fig. 2 (f).

3.1 *GaussianLens*: Cross-Modal Gaussian Densification Prediction Framework

Given initial Gaussians $\mathcal{G}^{init}$ and 2D images, our Gaussian densification framework *GaussianLens* (illustrated in Fig. 2) constructs initial multi-modal features, encodes them into latent features through a transformer network, which employs cross-attention to images to further fuse information, and finally decodes latent features into the parameters of new densified Gaussians $\mathcal{G}^{den}$. For brevity, below we present a simplified description. Please refer to Supp. Sec. B.1 for full details.

Multi-Modal Residual Gaussian Feature Initialization. As shown in Fig. 2 (a), given input images $\mathcal{I}$ and initial Gaussians $\mathcal{G}^{init}$, we first construct initial per-Gaussian features $\{g_j\}$ and image features $\{H_i\}$ by fusing 3D Gaussian and 2D multi-view information.

To associate input 3D Gaussians with 2D images, we render $\mathcal{G}^{init}$ at input views and compare them with the input images. Concretely, at each view i, we obtain reconstructed RGB, depth, and opacities $(\hat{I}_i, \hat{D}_i, \hat{A}_i)$, and explicitly compute the residual between groundtruth images and reconstruction as $E_i = I_i - \hat{I}_i$. We define the reconstruction-residual image features as $H_i^{recon} = (\hat{I}_i, \hat{D}_i, \hat{A}_i, E_i)$. We also extract dense multi-view features $\{H_i^{\text{mv}}\}_{i=1}^N$ with a pretrained multi-view feature extractor from [48]. Finally, we construct image features $\{H_i\}$ as $H_i = (I_i, H_i^{\text{mv}}, H_i^{recon})$.

We construct initial Gaussian features $\{g_j\}$ as $g_j = (g_j^{param}, g_j^{grad}, g_j^{proj})$, where g_j^{param} are the Gaussian parameters, g_j^{grad} are the gradients of the ren-

dering loss $\mathcal{L} = \sum_i ||E_i||^2$ with respect to the parameters, i.e., $\nabla_{G_j}\mathcal{L}$. We incorporate them to leverage the residual signal, similar to [6]. g_j^{proj} refers to local image features at the projected Gaussian centers. Concretely, we compute the 2D projection of Gaussian G_j's center to view i, take the bilinear interpolation of image feature H_i at the projection, and concatenate projection features from all views to obtain g_j^{proj}.

PointTransformer-based Feature Encoder with Projection-Based Cross-Attention. We encode initial Gaussian features $\{g_j\}$ into latent features $\{f_j\}$ with an encoder network ϕ_{enc} based on PointTransformerv3 (PTv3) [43], as shown in Fig. 2 (b). We adopt its standard U-Net architecture with serialized self-attention layers, progressive down-/up-sampling, and skip connections to efficiently extract spatial features at multiple scales.

We further employ a projection-based cross-attention layer (`ProjCrossAttn`) to fully integrate image information. As Fig. 2 (e) illustrates, given image features $\{H_i\}_{i=1}^N$ and projection matrices $\{\mathbf{P}_i\}_{i=1}^N$, for each Gaussian feature f_j centered at p_j, `ProjCrossAttn` projects the 3D Gaussian center p_j to each view i, bilinearly-interpolates image features H_i at the 2D projection to obtain sampled image feature $h_{i,j} = H_i[\pi_{\mathbf{P}_i}(p_j)]$. Finally, it applies a standard cross-attention with Gaussian feature f_j as the query and sampled image features $\{h_{i,j}\}_{i=1}^N$ as key and values. Formally,

$$\texttt{ProjCrossAttn}((p_j, f_j), \{H_i, \mathbf{P}_i\}_{i=1}^N) = \texttt{CrossAttn}(\{f_j\}, \{h_{i,j}\}_{i=1}^N), \quad (1)$$

where `CrossAttn` is a standard pair-wise cross-attention between token sets $\{f_j\}$ and$\{h_{i,j}\}_{i=1}^N$. This projection-based approach establishes localized correspondences and is more scalable than global cross-attention that relates all Gaussians to all image tokens [11].

To facilitate information exchange across varying scales, we build a multi-scale image feature pyramid $(\mathbf{H}, \mathbf{H}^{\frac{1}{2}}, \mathbf{H}^{\frac{1}{4}})$, and apply `ProjCrossAttn` to the last three decoder blocks of PTv3, fusing low-resolution image features with downsampled, low-resolution Gaussians and, conversely, high-resolution image features with full-resolution Gaussians.

In summary, ϕ_{enc} processes Gaussian features $\{(\mu_j, g_j)\}_{j=1}^M$ and image features $\mathbf{H}$ with spatial self-attention and multi-scale projection-based cross-attention to produce per-Gaussian features $\{f_j\}_{j=1}^M$.

Gaussian Densification Decoder. As shown in Fig. 2 (c), for each initial Gaussian G_j, we use an MLP decoder ϕ_{dec} to map its final feature f_j into the Gaussian parameters of K final densified Gaussians $\{\hat{G}_j^k\}_{k=1}^K$, expressed as residuals to the original Gaussian parameters. Formally,

$$\hat{G}_j^k = G_j + \Delta\hat{G}_j^k, \quad \{\Delta\hat{G}_j^k\}_{k=1}^K = \{(\Delta\boldsymbol{\mu}_j^k, \Delta\alpha_j^k, \Delta\boldsymbol{\Sigma}_j^k, \Delta\boldsymbol{c}_j^k)\}_{k=1}^K = \phi_{dec}(f_j).$$

K is a predefined densification factor, which we find sufficient to set to $K = 1$. [1] The final output of *GaussianLens* is the union of all densified Gaussians, i.e., $\mathcal{G}^{den} = \{\hat{G}_j^k\}_{j=1...M,k=1...K}$.

We train our model with mean squared error (MSE) between images rendered from predicted Gaussians and the groundtruth at N' novel target test views $\{I_i\}_{i=1}^{N'}$, within RoI masks $\{M_i\}_{i=1}^{N'}$:

$$\mathcal{L}_{\text{MSE}} = \sum_{i=1}^{N'} \text{sum}(M_i \cdot ||I_i^{pred} - I_i^{GT}||_2^2) / \sum_{i=1}^{N'} \text{sum}(M_i)$$

3.2 Pixel-Guided Gaussian Densification

Conceptually, *GaussianLens* densifies initial Gaussians $\mathcal{G}^{init}$ with learned cloning and refinement. However, for large resolution increases, solely cloning existing Gaussians is insufficient. For a 4× zoom-in, a single initial Gaussian becomes responsible for capturing a 4×4-pixel region. Aggregating all information and mapping them to Gaussian parameters is a challenging learning task.

To address the challenge, we propose pixel-guided densification, a mechanism that directly injects information from the high-resolution observations by augmenting the initial Gaussians with a set of **pixel-guided Gaussians**, $\mathcal{G}^{\text{pixel}} = \{G_{i,xy}\}_{1 \le i \le N, M_{i,xy}=1}$. As shown in Fig. 2 (e), we spawn one new Gaussian $G_{i,xy}$ for each pixel $\mathbf{p}_{i,xy}$ within the RoI mask M_i of each view i. The color $\boldsymbol{c}_{i,xy}$ of the Gaussian is initialized to the corresponding pixel color $I_{i,xy}$. The 3D position $\boldsymbol{\mu}_{i,xy}$ is determined by back-projecting the pixel along its camera ray to the depth rendered from the coarse initial 3DGS reconstruction. The opacity and scale are initialized to small constant values. Pixel-guided Gaussians explicitly incorporate dense appearance details, providing *GaussianLens* with a better foundation. We pass the union of input RoI Gaussians and pixel-guided Gaussians to *GaussianLens* for further densification and refinement, i.e., $\mathcal{G}^{init} \leftarrow \mathcal{G}^{RoI} \cup \mathcal{G}^{pixel}, \mathcal{G}^{den} = \text{GaussianLens}(\mathcal{G}^{init})$.

4 Experiment

4.1 Region-of-Interest View Synthesis Benchmark

Dataset. We build our region-of-interest view synthesis benchmark based on RealEstate10K (RE10K) [63] and DL3DV [23]. RE10K contains real estate walk-through videos captured at 1280×720 resolution, while most feed-forward Gaussian works [4, 36, 62] use a downscaled 256×256 version, with few [47] using higher resolution for qualitative results. DL3DV captures diverse, challenging scenes with 3840×2160 videos. So far, existing novel-view synthesis works [5, 13, 16, 23, 31, 47, 51] only use resolutions up to 960×540.

[1] To reduce redundancy and support selective densification, we can optionally predict an existence mask for each densified Gaussian. Please see Supp. Sec. A.3 for more details.

Zoom-in Resolution Setting We use 256×256 for low-resolution, full-sized input images for global capture on both datasets. For high-resolution images of the region-of-interest, we use 512×512 and 1024×1024 for RE10K and DL3DV, respectively. Since we focus on local regions that occupy a small portion of the full-sized image, to reduce redundancy, we use 256×256 crops from high-res images that enclose the RoI. This can also be viewed as mimicking close-up captures with available high-res, global-scale images in the dataset. Below, we refer to the two settings by "RE10K, $256 \rightarrow 512$ ($2\times$)" and "DL3DV, $256 \rightarrow 1024$ ($4\times$)" to emphasize the resolution increase and zoom-in factors.

RoI Generation To mimic the use case where users specify 3D RoIs via a 2D selection interface, we generate 3D RoIs by sampling 2D crops from context views and backprojecting them to the 3D scene. Please refer to Supp. Sec. B.4 for more details.

Evaluation Setting We use the official train/test scene splits and follow the context/target view selection protocols of pixelSplat [2] on RE10K and DepthSplat [47] on DL3DV. We use the same views for both low-resolution and high-resolution images. We evaluate the results using standard novel view synthesis metrics PSNR, SSIM [41], and LPIPS [60]. We adapt them to the Region-of-Interest setting and only apply them to pixels within the RoI mask. We also report efficiency metrics, including the number of predicted Gaussians, trainable model parameters, and per-iteration training time and memory consumption.

4.2 Baselines

We compare to the following adaptations of state-of-the-art feed-forward Gaussian reconstruction works DepthSplat [47], pixelSplat [2], and MVSplat [4]. 1) 'low-res full' is the standard model operating on 256×256 low-resolution, full-size inputs and predicts per-pixel Gaussians. We use this variant of DepthSplat to generate the initial Gaussians as inputs to our model. 2) 'high-res full' uses high-resolution, full-size inputs (512×512 on RE10K), and outputs per-pixel Gaussians at high-resolution. This variant has unfair access to additional information from the full-sized, high-resolution context images, and is therefore not directly comparable. 3) 'high-res crop' takes 256×256 RoI crops from full-size high-resolution images as inputs, and only outputs Gaussians for pixels in the crop, avoiding the costly per-high-res-pixel prediction. Please refer to Supp. Sec. B.3 for details.

As all baselines predict new Gaussians of the RoI independent of the global initial ones, they do not produce a single, consistent reconstruction with both global content and local details. We only use them for reference in evaluating the reconstruction quality of the RoI.

Table 1: Quantitative comparisons on DL3DV and RE10K. Best overall results are in **bold**. Best results under fair comparison, which excludes methods with privileged access to full high-res images (*), are underlined. Time and memory consumptions are measured with a batch size of 1 on an NVIDIA a6000 GPU.

	Method	PSNR↑	SSIM↑	LPIPS↓	Trainable Param.↓	Training Cost Time↓	Training Cost Mem.↓	Num. GS↓
RE10K 256→512	pixelSplat low-res full	25.94	0.820	0.128	118M	0.89s	14.32G	393K
	MVSplat low-res full	26.08	0.832	0.089	12M	0.49s	8.27G	131K
	DepthSplat low-res full	27.28	0.852	0.101	120M	0.66s	8.90G	131K
	MVSplat high-res full*	27.12	0.862	0.085	12M	1.15s	25.91G	524K
	DepthSplat high-res full*	28.26	0.875	0.085	120M	2.03s	32.66G	524K
	DepthSplat high-res crop	24.38	0.759	0.161	120M	0.78s	8.90G	131K
	Ours	**28.54**	**0.876**	**0.084**	43M	1.74s	13.27G	214K
DL3DV 256→1024	DepthSplat low-res full	22.31	0.652	0.286	120M	0.91s	8.15G	131K
	DepthSplat high-res full*	Out of Memory on an 80GB H100 GPU						
	DepthSplat high-res crop	19.51	0.571	0.352	120M	0.87s	8.10G	131K
	Ours	**23.74**	**0.721**	**0.225**	43M	1.67s	9.46G	220K

4.3 Benchmark Comparisons

Table 1 shows the quantitative comparison with baselines. Our method consistently improves upon initial Gaussians predicted by 'DepthSplat low-res full' by more than 1dB PSNR, effectively leveraging high-resolution observations. Compared to 'Depthsplat high-res full' with unfair access to full-size high-resolution images, we achieve on-par performance on RE10K with only 40% Gaussian budgets, fewer model parameters, shorter training time, and much smaller memory consumption. Our method's efficiency and reduced computational requirements are more evident in the DL3DV 256 → 1024 setting, where 'DepthSplat high-res full' fails to train as it runs out of memory even on an 80GB H100 GPU, revealing the inefficiency and lack of scalability of standard, uniform-resolution feed-forward models. 'DepthSplat high-res crop' scales to high resolutions by operating on crops. However, it struggles to learn reliable multi-view geometry from the small-overlap, off-center local image crops, leading to lower performance. In contrast, our method effectively leverages initial Gaussians as a coarse 3D scaffold to aggregate information onto.

As shown in Fig. 3, our model improves details originally blurred out in low-resolution initial reconstructions, such as thin structures and fine patterns. Please see Supp. Sec. C for more qualitative results.

4.4 Zero-Shot Generalization to Different Gaussian Sources

Despite trained exclusively on input 3D Gaussians predicted by DepthSplat, our method can zero-shot generalize to input 3D Gaussians from distinct sources: Gaussians predicted by unseen feed-forward models pixelSplat and MVSplat on

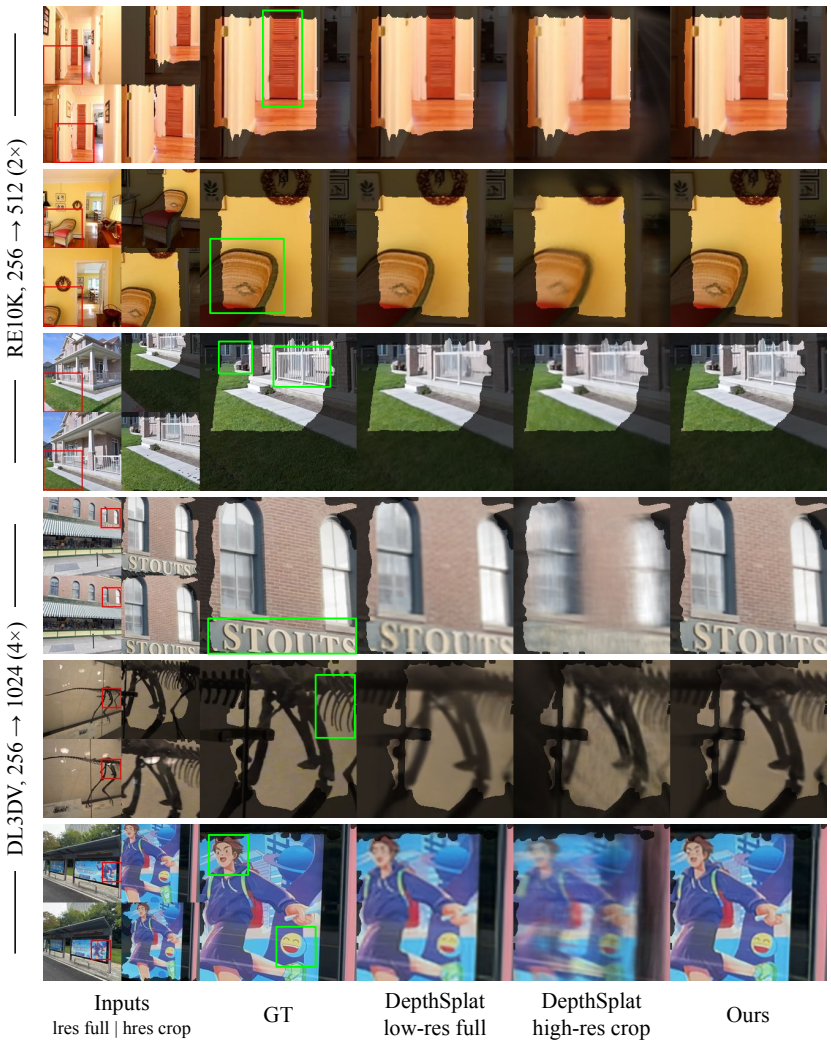

Fig. 3: Novel view synthesis on RealEstate10K [63] and DL3DV [23]. Our method reconstructs finer details by effectively leveraging high-resolution observations and initial Gaussians.

RE10K, and Gaussians per-scene optimized from dense observations on DL3DV. As summarized in Tab. 2, our model achieves consistent improvement upon the input Gaussians in all metrics, showing strong generalization ability to inputs from different distributions. This is crucial for the flexible deployment of the method and underscores the advantage of not relying on prior knowledge or access to the internal features of the Gaussian source. Figure 4 visualizes our improvements given DepthSplat-predicted and per-scene-optimized initial

Table 2: Generalization to different input Gaussian sources. We take our models trained exclusively on input Gaussians predicted by the DepthSplat low-res full model, and directly apply them to Gaussians from unseen feed-forward models pixelSplat and MVSplat on RE10K, or per-scene optimized Gaussians on DL3DV.

Dataset	Source of input Gaussians	input Gaussians			predicted densification		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
RE10K $_{256 \to 512}$	DepthSplat low-res full	27.28	0.852	0.101	28.54(+1.26)	0.876(+0.024)	0.084(-0.017)
	pixelSplat low-res full	25.94	0.820	0.128	27.09(+1.15)	0.844(+0.024)	0.108(-0.020)
	MVSplat low-res full	26.08	0.832	0.089	27.32(+1.24)	0.853(+0.021)	0.087(-0.002)
DL3DV $_{256 \to 1024}$	DepthSplat low-res full	22.31	0.652	0.286	23.74(+1.43)	0.721(+0.069)	0.225(-0.061)
	per-scene optim.	22.34	0.659	0.291	24.46(+2.12)	0.742(+0.083)	0.225(-0.066)

Gaussians. Our model improves detail reconstructions in both cases. Notably, as shown in row 3-4, when per-scene optimization provides a better initialization, our model effectively capitalizes on that advantage and produce more accurate final reconstructions. Please refer to Supp. Sec. A.1 for details.

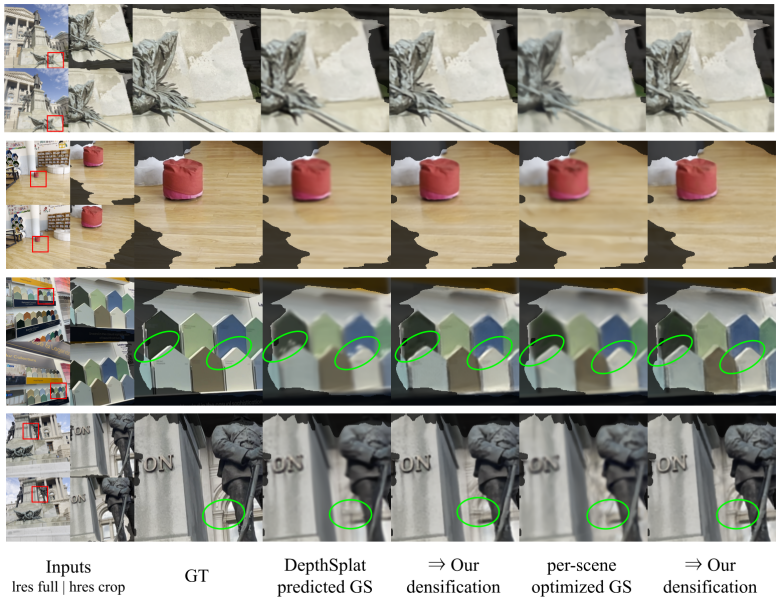

Fig. 4: Densification results given input Gaussians from DepthSplat prediction or per-scene optimization. Our model achieves zero-shot generalization to per-scene optimized 3D Gaussians and reconstructs better details in all cases. When per-scene optimization provides a better initialization (row 3-4), our model can amplify this advantage and produce more accurate final reconstructions.

4.5 Ablation Studies

We ablate our method under the DL3DV 256 → 1024 setting and summarize the results in Tab. 3.

Table 3: Ablation studies. All models are trained and evaluated under the DL3DV 256 → 1024 setting.

GS Source	PSNR↑	SSIM↑	LPIPS↓
input,$_{4\times}$	$23.27_{+0.96}$	0.695	0.260
input,$_{16\times}$	$23.25_{+0.94}$	0.697	0.260
pixel	$22.97_{+0.66}$	0.694	0.248
Ours (both)	$23.74_{+1.43}$	0.721	0.225

(a) Source Gaussians to densify from. 'input, $_{K\times}$' only densifies the original Gaussians $\mathcal{G}^{RoI}$ by densification factor K(K= 4 or 16). 'pixel' only densifies pixel-guided Gaussians $\mathcal{G}^{pixel}$. Ours densifies both sources of Gaussians $\mathcal{G}^{RoI} \cup \mathcal{G}^{pixel}$.

Initial Feature	PSNR↑	SSIM↑	LPIPS↓
no gradient	$23.58_{+1.27}$	0.713	0.233
no recon.	$23.31_{+1.00}$	0.705	0.238
no multi-view	$23.53_{+1.22}$	0.710	0.235
Ours	$23.74_{+1.43}$	0.721	0.225

(b) Initial features. 'no gradient' removes gradient feature g^{grad} from initial Gaussian features. 'no recon.', and 'no multi-view' remove reconstruction features H^{recon} and multi-view features H^{mv} from initial image features.

Attention Type	PSNR↑	SSIM↑	LPIPS↓
no attention	$22.65_{+0.34}$	0.664	0.273
global	$22.64_{+0.33}$	0.663	0.274
last block	$23.59_{+1.28}$	0.716	0.230
Ours	$23.74_{+1.43}$	0.721	0.225

(c) Image-Gaussian cross-attention. 'no attention' removes cross-attention. 'global' performs attention between all images and points at the bottleneck. 'last block' only performs projection-based cross-attention at the last block.

Source Gaussians to Densify from. Our method densifies Gaussians from both the input ($\mathcal{G}^{RoI}$) and pixel-guided densification ($\mathcal{G}^{pixel}$). As shown in Tab. 3a, densifying from either input Gaussians only ('input, $_{K\times}$'), or pixel-guided Gaussians only ('pixel') leads to decreased performance. Simply increasing the densification factor K from 4 to 16 does not lead to further improvement, implying the limitation of using input Gaussians only and the necessity of pixel-guided densification. Figure 5 presents a breakdown visualization of source Gaussians and their predicted updates. The network learns to update both input and pixel-guided Gaussians to collaboratively reconstruct the scene. While some details emerge from the input Gaussians, they serve more as a coarse backdrop, on which pixel-guided Gaussians render sharp details.

Multi-Modal Residual Feature Initialization. To analyze the construction of the initial Gaussian and image features, we remove gradient-based features g^{grad} from initial Gaussian features ('no gradient'), reconstruction-residual features H^{recon} or multi-view features H^{mv} from image features ('no recon.' and 'no multi-view'). As shown in Tab. 3b, all features contribute to the final performance. The most significant drop occurs when we remove image reconstructions

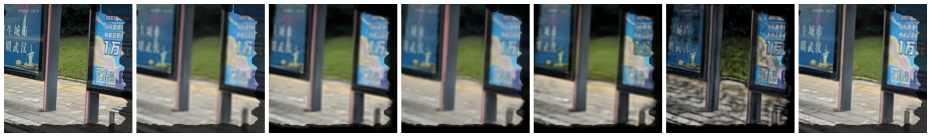

Fig. 5: Breakdown visualization of source and updated Gaussians. Our network takes input Gaussians in the RoI (c) and pixel-guided Gaussians (d), and predicts their updates (e, f). While some details emerge in updated input Gaussians (e), pixel-guided ones (f) reconstruct sharper details.

and residuals rendered from initial Gaussians ('no recon.'), highlighting the importance of explicitly associating Gaussians and images through rendering for effective residual learning.

Projection-Based Gaussian-Image Cross-attention. We replace our multi-scale, projection-based cross-attention with 1) no cross-attention with images ('no attention'), PTv3 performs self-attention only; 2) a global cross-attention between all image and Gaussian tokens ('global'), similar to [11], which is only applied to the bottleneck block with downsampled tokens due to the squared computation complexity; 3) a single-scale projection-based cross-attention at the last transformer block ('last block'). As shown in Tab. 3c, both no attention and global attention suffer a significant performance drop, suggesting the vital role of local image information in Gaussian densification prediction. Multi-scale attention further improves performance upon single-scale ('last block').

5 Conclusion

In this paper, we introduce the task of localized high-resolution reconstruction via on-demand Gaussian densification. Our formulation addresses the practical need for spatially varying detail reconstruction, avoids the prohibitive cost of uniformly high-resolution reconstruction, and enables the effective use of high-resolution observations. We develop *GaussianLens*, a generalizable densification prediction framework that effectively fuses multi-modal input information. We further propose a novel pixel-guided densification mechanism to capture details under significant resolution increases. Our model achieves state-of-the-art performance in localized high-resolution reconstruction while consuming fewer computational resources.

Limitations and Future Directions Our model focuses on improving detail reconstruction based on a coarse initial Gaussian reconstruction. It is not designed to handle catastrophic errors already present in the initial reconstruction. Developing a method to leverage emerging geometry cues from high-resolution observations to recover from reconstruction failures at low resolution would be an exciting future direction.

References

1. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023)
2. Charatan, D., Li, S.L., Tagliasacchi, A., Sitzmann, V.: pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
3. Chen, A., Xu, H., Esposito, S., Tang, S., Geiger, A.: Lara: Efficient large-baseline radiance fields. In: European Conference on Computer Vision (2025)
4. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision (2025)
5. Chen, Y., Zheng, C., Xu, H., Zhuang, B., Vedaldi, A., Cham, T.J., Cai, J.: Mvsplat360: Feed-forward 360 scene synthesis from sparse views. arXiv preprint arXiv:2411.04924 (2024)
6. Chen, Y., Wang, J., Yang, Z., Manivasagam, S., Urtasun, R.: G3r: Gradient guided generalizable reconstruction. In: European Conference on Computer Vision. pp. 305–323. Springer (2024)
7. Chen, Y., Mihajlovic, M., Chen, X., Wang, Y., Prokudin, S., Tang, S.: Splatformer: Point transformer for robust 3d gaussian splatting. arXiv preprint arXiv:2411.06390 (2024)
8. Cheng, K., Long, X., Yang, K., Yao, Y., Yin, W., Ma, Y., Wang, W., Chen, X.: Gaussianpro: 3d gaussian splatting with progressive propagation. In: Forty-first International Conference on Machine Learning (2024)
9. Di Sario, F., Renzulli, R., Grangetto, M., Sugimoto, A., Tartaglione, E.: Gode: Gaussians on demand for progressive level of detail and scalable compression. arXiv preprint arXiv:2501.13558 (2025)
10. Fan, Z., Cong, W., Wen, K., Wang, K., Zhang, J., Ding, X., Xu, D., Ivanovic, B., Pavone, M., Pavlakos, G., et al.: Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds. arXiv preprint arXiv:2403.20309 **2**(3), 4 (2024)
11. Fan, Z., Zhang, J., Cong, W., Wang, P., Li, R., Wen, K., Zhou, S., Kadambi, A., Wang, Z., Xu, D., et al.: Large spatial model: End-to-end unposed images to semantic 3d. Advances in neural information processing systems **37**, 40212–40229 (2024)
12. Feng, X., He, Y., Wang, Y., Yang, Y., Li, W., Chen, Y., Kuang, Z., Fan, J., Jun, Y., et al.: Srgs: Super-resolution 3d gaussian splatting. arXiv preprint arXiv:2404.10318 (2024)
13. Fischer, T., Bulò, S.R., Yang, Y.H., Keetha, N.V., Porzi, L., Müller, N., Schwarz, K., Luiten, J., Pollefeys, M., Kontschieder, P.: Flowr: Flowing from sparse to dense 3d reconstructions. arXiv preprint arXiv:2504.01647 (2025)
14. Hong, S., Jung, J., Shin, H., Han, J., Yang, J., Luo, C., Kim, S.: Pf3plat: Pose-free feed-forward 3d gaussian splatting. arXiv preprint arXiv:2410.22128 (2024)
15. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. In: SIGGRAPH (2024)
16. Kang, G., Yoo, J., Park, J., Nam, S., Im, H., Shin, S., Kim, S., Park, E.: Selfsplat: Pose-free and 3d prior-free generalizable 3d gaussian splatting. arXiv preprint arXiv:2411.17190 (2024)

17. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering (2023)
18. Kerbl, B., Meuleman, A., Kopanas, G., Wimmer, M., Lanvin, A., Drettakis, G.: A hierarchical 3d gaussian representation for real-time rendering of very large datasets. ACM Transactions on Graphics (TOG) **43**(4), 1–15 (2024)
19. Kheradmand, S., Rebain, D., Sharma, G., Sun, W., Tseng, Y.C., Isack, H., Kar, A., Tagliasacchi, A., Yi, K.M.: 3d gaussian splatting as markov chain monte carlo. Advances in Neural Information Processing Systems **37**, 80965–80986 (2024)
20. Lee, J.L., Li, C., Lee, G.H.: Disr-nerf: Diffusion-guided view-consistent super-resolution nerf. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20561–20570 (2024)
21. Leroy, V., Cabon, Y., Revaud, J.: Grounding image matching in 3d with mast3r. In: European Conference on Computer Vision. pp. 71–91. Springer (2024)
22. Li, Y., Wang, J., Chu, L., Li, X., Kao, S.h., Chen, Y.C., Lu, Y.: Streamgs: Online generalizable gaussian splatting reconstruction for unposed image streams. arXiv preprint arXiv:2503.06235 (2025)
23. Ling, L., Sheng, Y., Tu, Z., Zhao, W., Xin, C., Wan, K., Yu, L., Guo, Q., Yu, Z., Lu, Y., et al.: Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22160–22169 (2024)
24. Liu, T., Wang, G., Hu, S., Shen, L., Ye, X., Zang, Y., Cao, Z., Li, W., Liu, Z.: Mvsgaussian: Fast generalizable gaussian splatting reconstruction from multi-view stereo. In: European Conference on Computer Vision (2025)
25. Liu, Y., Luo, C., Fan, L., Wang, N., Peng, J., Zhang, Z.: Citygaussian: Real-time high-quality large-scale scene rendering with gaussians. In: European Conference on Computer Vision. pp. 265–282. Springer (2024)
26. Lu, L., Gao, H., Dai, T., Zha, Y., Hou, Z., Wu, J., Xia, S.T.: Large point-to-gaussian model for image-to-3d generation. In: Proceedings of the ACM International Conference on Multimedia (2024)
27. Lyu, Y., Cheng, K., Kang, X., Chen, X.: Resgs: Residual densification of 3d gaussian for efficient detail recovery. arXiv preprint arXiv:2412.07494 (2024)
28. Nam, S., Sun, X., Kang, G., Lee, Y., Oh, S., Park, E.: Generative densification: Learning to densify gaussians for high-fidelity generalizable 3d reconstruction. arXiv preprint arXiv:2412.06234 (2024)
29. Ren, K., Jiang, L., Lu, T., Yu, M., Xu, L., Ni, Z., Dai, B.: Octree-gs: Towards consistent real-time rendering with lod-structured 3d gaussians. arXiv preprint arXiv:2403.17898 (2024)
30. Rota Bulò, S., Porzi, L., Kontschieder, P.: Revising densification in gaussian splatting. In: European Conference on Computer Vision. pp. 347–362. Springer (2024)
31. Seo, Y., Choi, Y.S., Son, H.S., Uh, Y.: Flod: Integrating flexible level of detail into 3d gaussian splatting for customizable rendering (2024), https://arxiv.org/abs/2408.12894
32. Shen, Q., Wu, Z., Yi, X., Zhou, P., Zhang, H., Yan, S., Wang, X.: Gamba: Marry gaussian splatting with mamba for single view 3d reconstruction. arXiv preprint arXiv:2403.18795 (2024)
33. Shen, Y., Ceylan, D., Guerrero, P., Xu, Z., Mitra, N.J., Wang, S., Frühstück, A.: Supergaussian: Repurposing video models for 3d super resolution. In: European Conference on Computer Vision. pp. 215–233. Springer (2024)
34. Shi, Y., Morin, G., Gasparini, S., Ooi, W.T.: Lapisgs: Layered progressive 3d gaussian splatting for adaptive streaming. arXiv preprint arXiv:2408.14823 (2024)

35. Szymanowicz, S., Insafutdinov, E., Zheng, C., Campbell, D., Henriques, J.F., Rupprecht, C., Vedaldi, A.: Flash3d: Feed-forward generalisable 3d scene reconstruction from a single image. arXiv preprint arXiv:2406.04343 (2024)
36. Szymanowicz, S., Rupprecht, C., Vedaldi, A.: Splatter image: Ultra-fast single-view 3d reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
37. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In: European Conference on Computer Vision (2025)
38. Tang, S., Ye, W., Ye, P., Lin, W., Zhou, Y., Chen, T., Ouyang, W.: Hisplat: Hierarchical 3d gaussian splatting for generalizable sparse-view reconstruction. arXiv preprint arXiv:2410.06245 (2024)
39. Wan, Y., Shao, M., Cheng, Y., Zuo, W.: S2gaussian: Sparse-view super-resolution 3d gaussian splatting. arXiv preprint arXiv:2503.04314 (2025)
40. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20697–20709 (2024)
41. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
42. Wewer, C., Raj, K., Ilg, E., Schiele, B., Lenssen, J.E.: latentsplat: Autoencoding variational gaussians for fast generalizable 3d reconstruction. arXiv preprint arXiv:2403.16292 (2024)
43. Wu, X., Jiang, L., Wang, P.S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H.: Point transformer v3: Simpler faster stronger. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
44. Xia, J., Liu, L.: Close-up-gs: Enhancing close-up view synthesis in 3d gaussian splatting with progressive self-training. arXiv preprint arXiv:2503.09396 (2025)
45. Xia, J., Sun, L., Liu, L.: Enhancing close-up novel view synthesis via pseudo-labeling. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 8567–8574 (2025)
46. Xie, S., Wang, Z., Zhu, Y., Pan, C.: Supergs: Super-resolution 3d gaussian splatting via latent feature field and gradient-guided splitting. arXiv preprint arXiv:2410.02571 (2024)
47. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depthsplat: Connecting gaussian splatting and depth. arXiv preprint arXiv:2410.13862 (2024)
48. Xu, H., Zhang, J., Cai, J., Rezatofighi, H., Yu, F., Tao, D., Geiger, A.: Unifying flow, stereo and depth estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(11), 13941–13958 (2023)
49. Xu, Y., Shi, Z., Yifan, W., Chen, H., Yang, C., Peng, S., Shen, Y., Wetzstein, G.: Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. arXiv preprint arXiv:2403.14621 (2024)
50. Yan, Z., Low, W.F., Chen, Y., Lee, G.H.: Multi-scale 3d gaussian splatting for anti-aliased rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20923–20931 (2024)
51. Ye, B., Liu, S., Xu, H., Li, X., Pollefeys, M., Yang, M.H., Peng, S.: No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. arXiv preprint arXiv:2410.24207 (2024)

52. Ye, Z., Li, W., Liu, S., Qiao, P., Dou, Y.: Absgs: Recovering fine details in 3d gaussian splatting. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 1053–1061 (2024)
53. Yoon, Y., Yoon, K.J.: Cross-guided optimization of radiance fields with multi-view image super-resolution for high-resolution novel view synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12428–12438 (2023)
54. Yu, X., Zhu, H., He, T., Chen, Z.: Gaussiansr: 3d gaussian super-resolution with 2d diffusion priors. arXiv preprint arXiv:2406.10111 (2024)
55. Yu, Z., Sattler, T., Geiger, A.: Gaussian opacity fields: Efficient and compact surface reconstruction in unbounded scenes. arXiv preprint arXiv:2404.10772 (2024)
56. Zhang, C., Xu, H., Wu, Q., Gambardella, C.C., Phung, D., Cai, J.: Pansplat: 4k panorama synthesis with feed-forward gaussian splatting. arXiv preprint arXiv:2412.12096 (2024)
57. Zhang, C., Zou, Y., Li, Z., Yi, M., Wang, H.: Transplat: Generalizable 3d gaussian splatting from sparse multi-view images with transformers. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 9869–9877 (2025)
58. Zhang, C., Song, H., Wei, Y., Chen, Y., Lu, J., Tang, Y.: Geolrm: Geometry-aware large reconstruction model for high-quality 3d gaussian generation. arXiv preprint arXiv:2406.15333 (2024)
59. Zhang, K., Bi, S., Tan, H., Xiangli, Y., Zhao, N., Sunkavalli, K., Xu, Z.: Gs-lrm: Large reconstruction model for 3d gaussian splatting. In: European Conference on Computer Vision (2025)
60. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)
61. Zhang, X., Pan, W., Bao, C., Zhang, X., Xiang, X., Jiang, H., Bao, H.: Lookcloser: Frequency-aware radiance field for tiny-detail scene. arXiv preprint arXiv:2503.18513 (2025)
62. Zhang, Z., Hu, W., Lao, Y., He, T., Zhao, H.: Pixel-gs: Density control with pixel-aware gradient for 3d gaussian splatting. arXiv preprint arXiv:2403.15530 (2024)
63. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. arXiv preprint arXiv:1805.09817 (2018)
64. Zou, Z.X., Yu, Z., Guo, Y.C., Li, Y., Liang, D., Cao, Y.P., Zhang, S.H.: Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)