StreamGS: Online Generalizable Gaussian Splatting Reconstruction for Unposed Image Streams

Yang Li[1,4,*] Jinglu Wang[1] Lei Chu[1] Xiao Li[1]
Shiu-Hong Kao[1,2,*] Ying-Cong Chen[2,3] Yan Lu[1]
[1]Microsoft Research Asia [2]HKUST [3]HKUST(GZ) [4] Rutgers University
yangliaftermath@gmail.com skao@cse.ust.hk yingcongchen@ust.hk
{jinglu.wang, leichu, li.xiao, yanlu}@microsoft.com

Abstract

The advent of 3D Gaussian Splatting (3DGS) has advanced 3D scene reconstruction and novel view synthesis. With the growing interest of interactive applications that need immediate feedback, online 3DGS reconstruction in real-time is in high demand. However, none of existing methods yet meet the demand due to three main challenges: the absence of predetermined camera parameters, the need for generalizable 3DGS optimization, and the necessity of reducing redundancy. We propose ***StreamGS****, an online generalizable 3DGS reconstruction method for unposed image streams, which progressively transform image streams to 3D Gaussian streams by predicting and aggregating per-frame Gaussians. Our method overcomes the limitation of the initial point reconstruction [27] in tackling out-of-domain (OOD) issues by introducing a content adaptive refinement. The refinement enhances cross-frame consistency by establishing reliable pixel correspondences between adjacent frames. Such correspondences further aid in merging redundant Gaussians through cross-frame feature aggregation. The density of Gaussians is thereby reduced, empowering online reconstruction by significantly lowering computational and memory costs. Extensive experiments on diverse datasets have demonstrated that StreamGS achieves quality on par with optimization-based approaches but does so 150 times faster, and exhibits superior generalizability in handling OOD scenes.*

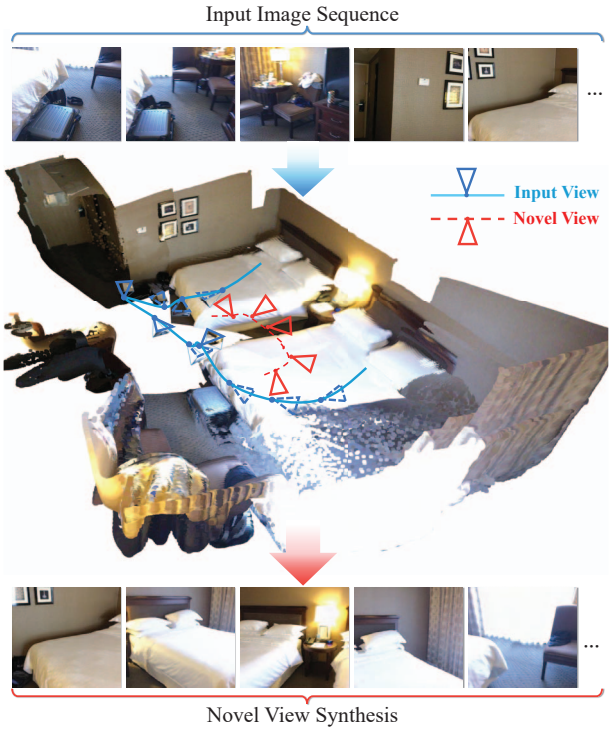

Figure 1. The proposed StreamGS efficiently transforms image streams into Gaussian streams by progressively reconstructing and aggregating per-frame 3D Gaussians. We show our reconstructed 3DGS (visualized as points) alongside estimated camera poses (in blue), and synthesized novel views.

1. Introduction

The field of 3D Scene reconstruction [10, 30] for novel view synthesis from image streams has gained increasing attention, due to its significance in enabling interactive applications that offer users instant feedback. In this context, the advent of 3D Gaussian Splatting (3DGS) [12] marks a major advancement in high-quality, real-time rendering. This progress highlights the urgency for efficient, on-the-fly generation of 3DGS from image streams.

Nonetheless, online 3DGS reconstruction faces unique challenges. (1) **Unknown camera.** The conventional preprocessing using Structure from Motion (SfM) [21] for camera estimation is impractical for real-time streaming.

*Work done during the internship at MSRA.

This is due to the absence of the full image set and the time-consuming computation. Thus, there is a need for methods that can operate without pre-determined cameras. (2) **Generalizability.** 3DGS reconstruction requires multi-iteration optimization, impractical for online applications due to the need for all images in advance. This restricts the development of a generalizable method that can process image streams in a feed-forward manner. (3) **Redundancy.** The significant overlap between frames leads to high redundancy when reconstructing Gaussians individually for each frame, increasing the streaming process's resource intensity.

Scene reconstruction without known cameras has been explored in SLAM-based [30, 34, 35, 40, 41] and NeRF-based [1, 23] methods. Only a few 3DGS-based methods are discussed [9, 20]. These methods primarily consider camera poses as learnable parameters and optimized alongside Gaussians through the iterative optimization. Yet, this optimization-driven approach for aligning each frame considerably increases the reconstruction time, rendering it impractical in the online scenario.

Recently, generalizable 3DGS reconstruction has been investigated for sparse views [2, 3, 24]. These approaches transform pixels into Gaussians, whose parameters are decoded from images via 2D encoder-decoder networks. They achieve the 3DGS reconstruction for each image in a single feed-forward pass. However, these generalizable approaches are primarily designed for monocular or binocular settings, suited to sparse-view inputs. In addition, multi-view methods [2, 3] typically infer Gaussian centers through stereo matching, highly dependent on known cameras, limiting their applicability for online scenarios with numerous unposed images. Furthermore, they combine multi-view 3DGS sets by simply uniting them, which overlooks cross-view alignment and overlaps, leading to misalignment and redundancy issues within image streams. The recent method [7] performs Gaussian downsampling in 3D space to reduce redundancy, but it necessitates traversing and processing Gaussians in 3D grids, which is also time-consuming and is highly dependent on the grid resolution.

Emerging models like DUSt3R [27] and MASt3R [15] enable sparse-view geometry reconstruction by simultaneously predicting 3D points and estimating camera parameters. These advances pave the way for more efficient, feed-forward, pose-free 3D reconstruction methods. A straightforward approach to generalize 3DGS reconstruction is to add a Gaussian predictor to the DUSt3R-like framework. However, this introduces several challenges. These models usually require datasets with ground truth 3D geometry, which may not always be available. Additionally, applying pretrained models to out-of-domain (OOD) data can result in inaccurate pose and 3D point estimations. Moreover, generating 3D points for each frame individually causes redundancy due to overlapping adjacent frames, potentially leading to ghosting artifacts from pose estimation errors.

In this paper, we introduce **StreamGS**, a novel pipeline for online, generalizable 3DGS reconstruction from unposed image streams. StreamGS aims to progressively construct and update the 3DGS representation of the scene frame-by-frame, in a feed-forward manner, as illustrated in Fig. 1. We leverage the pretrained DUSt3R to initially predict 3D point for the current frame using the previous frame as a reference. However, this initialization may encounter inaccuracies due to OOD issues. To mitigate this, we capitalize the insight that adjacent frames offer sufficient correspondences to refine the reconstruction. Unlike DUSt3R using predicted 3D points to establish correspondences, we adopt content-adaptive descriptors for more reliable matching, allowing for the **adaptive refinement** of the reconstruction by enhancing consistency between adjacent views. Furthermore, such correspondences help to prune redundant pixel-aligned Gaussians. Correlated pixel-wise features across frames are effectively aggregated, removing duplicates and achieving **adaptive density control**. Finally, we decode Gaussians from such aggregated features. StreamGS is adept at predicting and integrating Gaussians for the current frame into the existing Gaussian set seamlessly with a feed-forward pass. In summary, our contribution is three-fold.

- We introduce a novel pipeline for the online, generalizable reconstruction of image streams without requiring camera parameters, marking a first in this field.
- The proposed adaptive refinement enhance cross-frame consistency of 3DGS reconstruction, and the adaptive density control mechanism minimizes adjacent-view redundancy, thereby highly reducing computational costs in online reconstruction.
- Upon evaluation across diverse datasets, our method achieves high novel view synthesis quality comparable to the optimization-based method [9] but with 150x faster reconstruction speed. Additionally, our method outperforms existing pose-dependent generalizable 3DGS methods in handling OOD scenes, showing superior generalizability.

2. Related Works

Generalizable 3D Gaussian Splatting. Many recent studies aim to propose generalizable 3D-GS methods capable of predicting Gaussians within a single feed-forward pass. These works can be classified into two main categories: single-view reconstruction and multi-view reconstruction. Single-view reconstruction does not involve pose estimation as there are no multi-view constraints. Inspired by the insight from LRM [11] that large transformer-based [26] backbone networks can learn 3D priors from large-scale 3D data, the potential of predicting Gaussians from a

single image in a single feed-forward pass has been comprehensively explored. Numerous feed-forward models have been proposed, such as GRM [29], TriplaneGS [42], and GMamba [22]. However, these methods are primarily not applicable to multi-view scenarios as they always assume canonical poses.

Concurrently, many works focusing on multi-view inputs follow a similar paradigm, such as GS-LRM [36], LGM [25], and MVGMamba [31]. They concatenate the input images with camera embeddings like Plücker rays to facilitate the network in learning the proper fusion of Gaussians from multi-view inputs. However, these large 3D backbone networks mostly perform well only on synthetic objects due to the shortage of large-scale scene-level 3D data in the real world. Referring to generalizable NeRF methods like pixelNeRF [32], other research turns to multi-view stereo (MVS) matching to locate or initialize the centers of Gaussians, with other attributes decoded using a lightweight 2D encoder. Representative works with this design include MVSGaussian [18], pixelSplat [2], and MVSplat [3]. However, both camera ray embedding in large transformer-based models and stereo matching rely on known poses and intrinsics of each input view. Another main limitation of these methods is that they focus on sparse-view inputs. PixelSplat [2] and MVSplat [3] only support up to three views. Therefore, existing generalizable 3D-GS models cannot address the problem of feed-forward reconstruction from endless image streams.

Pose-free 3D Gaussian Splatting. Recently, many works have aimed to eliminate the need for Structure-from-Motion (SfM) preprocessing steps using COLMAP [21] software. Following the design of previous pose-free NeRF methods like NoPe-NeRF [1], Lu-NeRF [4], and localRF [19], CF-3DGS [9] introduces depth priors into the optimization of 3D-GS and performs progressive reconstruction. As each new image arrives, CF-3DGS optimizes both the pose and 3D Gaussians of the input image based on its depths, aiming to align the 3DGS from the new image with the preceding reconstruction. However, it still relies on known camera intrinsics. CF-3DGS is also not robust, as the accuracy of depth priors significantly impacts its reconstruction quality, limiting its application to common scenes. Moreover, it depends on thousands of optimization steps for each view, significantly extending the reconstruction time for each scene. Compared to generalizable 3D-GS models, current pose-free methods are so inefficient that they cannot be applied to image streams with a large number of frames.

Online 3D reconstruction of image streams. Online 3D reconstruction has been extensively studied in the field of SLAM [30, 34, 35, 39–41]. However, these methods typically involve additional information and most leverage SDF and NeRF as scene representations. NICE-SLAM [39] uses RGB-D streams as input, with the reconstructed scene represented by NeRF. NICER-SLAM [40] relies on geometric priors, including surface normals and depths. Surfel-NeRF [10] focuses on the novel view synthesis quality of online reconstruction with RGB streams, but it requires the poses and intrinsics of frames. Gaussian-SLAM [34] reconstructs the scene using 3D-GS, but it also relies on RGB-D streams.

3. Methods

Given a sequence of *unposed* images over time, our objective is to progressively reconstruct the 3D Gaussian Splatting (3DGS) representation in an online manner. Specifically, at each timestamp t, the goal is to derive the 3DGS $\mathcal{G}^t$, which encapsulates the 3D scene aggregated from the images $\mathcal{I}^t = \{\mathbf{I}^i\}_{i=1}^t$.

Fig. 2 illustrates our overall framework. In order to make reconstruction efficient with limited computational resources, we employ an incremental construction strategy. At each time step t, we focus on generating the 3DGS $\mathbf{G}^t$ of the current frame and merge it with the previous accumulated reconstruction $\mathcal{G}^{t-1}$ to obtain the full reconstruction $\mathcal{G}^t$ of the current time step. Specifically, with current frame $\mathbf{I}^t$, we use $\mathbf{I}^{t-1}$ as reference frame and estimate the point maps and cameras of each using an *initial reconstruction* module. With newly established matches between current and reference views, we further refine the quality of the points and the cameras in the *adaptive refinement* module. Finally, we generate $\mathbf{G}^t$ using the refined points and features of $\mathbf{I}^t$, and merge it with previous reconstructions according to the established matches, achieving the *feed-forward adaptive density control (ADC)*.

3.1. Preliminaries

Gaussian splatting. Previous work [12] represents a scene or object using a set of Gaussian distributions. Specifically each gaussian primitive could be denoted as $G(x;\mu,\Sigma) = e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$ and the covariance Σ is decomposed into the rotation matrix R and scaling matrix S to ensure the positive semi-definiteness during optimization, that is $\Sigma = RSS^TR^T$. The view-dependent color of the appearance is represented by a set of spherical harmonics (SH) coeffients and oppacity value α.

3.2. Initial Two-view Reconstruction

Given the current frame $\mathbf{I}^t$ and a reference frame $\mathbf{I}^{t'}$, we use a coarse predictor ϕ_{3D} and estimate the point map $\mathbf{X}^{t|t'}$ of the current frame under the local coordinate system of the reference frame together with its corresponding confidence map $\mathbf{C}^{t|t'}$, where the superscript $\cdot|t'$ indicates that the local coordinate system adapts to that of $\mathbf{I}^{t'}$. Intuitively, the predicted point map stores the 3D point coordinate that the

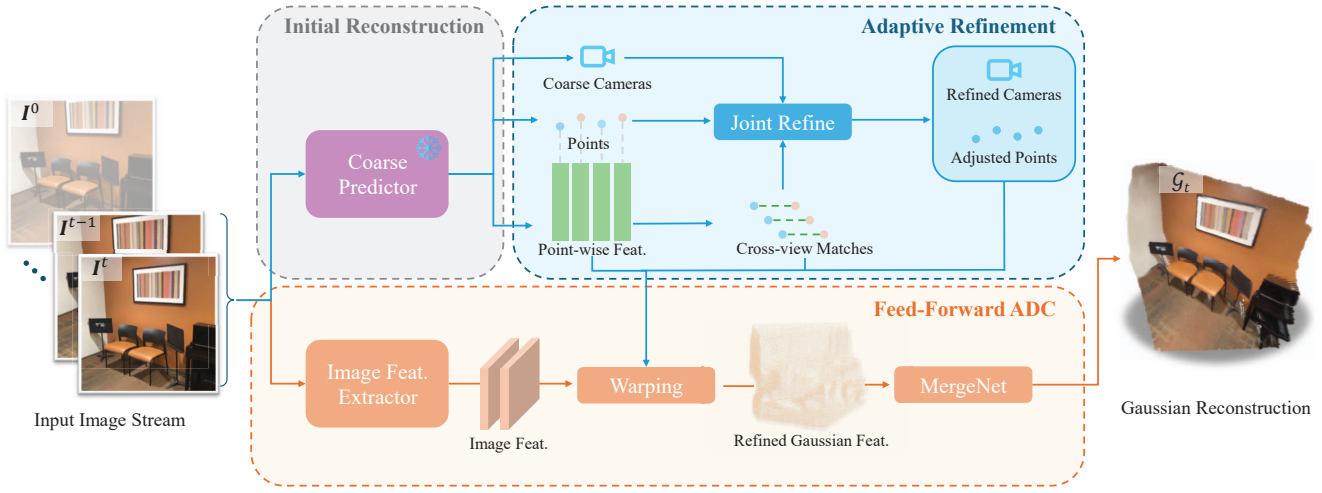

Figure 2. **Method overview.** Our StreamGS progressively reconstruct and aggregate 3D Gaussians from the unposed image stream. Given the adjacent image pair $(\mathbf{I}^{t-1}, \mathbf{I}^{t})$, we first perform the initial reconstruction that predicts pixel-wise 3D points with their features and coarse camera poses, using a pretrained coarse predictor. Since the coarse predictions may suffer from OOD issues, we refine both the camera poses and 3D positions by establishing new point-wise correspondences. We aggregate cross-frame image and 3D features by warping and merging to reduce redundancy. Finally we decode the aggregated features to Gaussian primitives.

pixel unprojected to in, and the confidence maps measure the certainty of point maps at each pixel, reflecting a prior-based estimation of reconstruction accuracy and difficulty. Formally, we define

$$(\mathbf{X}^{t|t'}, \mathbf{X}^{t'|t'}, \mathbf{C}^{t|t'}, \mathbf{C}^{t'|t'}) = \phi_{3D}(\mathbf{I}^{t}, \mathbf{I}^{t'}). \tag{1}$$

In order to help align the current local-coordinate pointmap with the global coordinate, we also get the output in the local coordinate system of $\mathbf{I}^t$ by computing $\phi_{3D}(\mathbf{I}^{t'}, \mathbf{I}^{t})$. With these predicted point maps, we could further estimate the camera matrix $\mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}]$, composed of its intrinsic parameters $\mathbf{K}$, and its extrinsic parameters, the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$. Specifically we assume that the principle points $\mathbf{c}$ are centered and pixels are squares and have

$$\hat{f}^t = \arg\min_{f} \Sigma_{\mathbf{p}\in\mathbf{I}^t} \parallel \mathbf{p} - \mathbf{c} - f\frac{(\mathbf{x_p}, \mathbf{y_p})}{\mathbf{z_p}} \parallel, \tag{2}$$

where $(\mathbf{x_p}, \mathbf{y_p}, \mathbf{z_p}) \in \mathbf{X}^{t|t}$ is the 3D point coordinate that the pixel $\mathbf{p}$ unprojected to. Moreover, we approximate relative pose $\mathbf{P}^t = [\mathbf{R}, \mathbf{t}]$ of $\mathbf{I}^t$ to $\mathbf{I}^{t'}$ by solving the following points registration problem:

$$\begin{aligned}[\mathbf{R}^t|\mathbf{t}^t] &= \arg\min_{s,\mathbf{R},\mathbf{t}} \sum_{\mathbf{p}\in\mathbf{I}^{t-1}} \mathbf{C}^t(\mathbf{p}) \parallel s(\mathbf{R}\mathbf{X}^{t'|t'}(\mathbf{p}) + \mathbf{t}) - \mathbf{X}^{t'|t}(\mathbf{p}) \parallel^2, \\ \mathbf{C}^t &= \mathbf{C}^{t'|t'} \odot \mathbf{C}^{t'|t},\end{aligned} \tag{3}$$

where $\odot$ is the Hadamard product and s is the scale factor.

In our implementation, we leverage DUSt3R [27] as the coarse predictor due to its effective pretraining on dedicated 3D scene datasets and its efficiency in reconstructing 3D points. For each time step t, we process the pair $(\mathbf{I}^{t-1}, \mathbf{I}^{t})$ to derive the 3D points $(\mathbf{X}^{t|t-1}, \mathbf{X}^{t-1|t-1})$ corresponding to both frames referenced in the coordinate frame $t-1$, the reversed pair $(\mathbf{I}^{t}, \mathbf{I}^{t-1})$ to obtain the same points $(\mathbf{X}^{t-1|t}, \mathbf{X}^{t|t})$ but in the coordinate frame of t, which will be reused at the next timestamp to boost efficiency. For simplicity, and without loss of generality, the following discussion will focus on the reconstruction of the image pair $(\mathbf{I}^{t-1}, \mathbf{I}^{t})$. Note that training ϕ_{3D} requires 3D geometric supervision, which may not be available in our monocular video input scenario. Thus, we leverage the pretrained ϕ_{3D} from [27].

3.3. Adaptive Refinement

We note that the initial reconstruction quality is compromised due to the OOD challenge as the coarse predictor is frozen. This observation motivates us to enhance reconstruction through content adaptation, with the goal of adaptively refining both poses and 3D reconstructions.

We perform the adaptive refinement based on establishing new robust matches between adjacent frames. We employ a matching head, ϕ_{match}, to extract local 3D features, denoted as $(\mathbf{F}^{t-1}_{3D}, \mathbf{F}^{t}_{3D}) \in \mathbb{R}^{H\times W\times d}$, from the consecutive image pair $(\mathbf{I}^{t-1}, \mathbf{I}^{t})$. The matches between the two images can be established through nearest reciprocal (NN) searching, satisfying the following condition:

$$\begin{aligned}\mathcal{M}^{t-1,t} = \{i_k \leftrightarrow j_k | i_k = \mathrm{NN}(j_k) \text{ and } j_k = \mathrm{NN}(i_k)\}_{k=1}^{N}, \\ \text{s.t. } \mathrm{NN}(i_k) = \arg\min_{0\le j_k\le H\times W} | 1 - \cos < \mathbf{F}^{t-1}_{3D,i_k}, \mathbf{F}^{t}_{3D,j_k} >|,\end{aligned} \tag{4}$$

where i_k, j_k are pixel index in image $\mathbf{I}^{t-1}, \mathbf{I}^{t}$ respectively, and the measurement of feature distance is cosine similarity. With the correspondences found, a residual transform

$\mathbf{\Delta} = [\Delta\mathbf{R}, \Delta\mathbf{t}]$ can be re-estimated following Eq. (3) by only taking matched points into account. Then we apply the residual transform to the pointmap $\mathbf{X}^{t,t-1}$ to retain refined $\tilde{\mathbf{X}}^{t|t-1}$, and the pose of $\mathbf{I}^t$ is updated to $\tilde{\mathbf{P}}^t$ by performing PnP-RANSAC [8, 14] on 3D-2D correspondences derived from matches.

Gaussian decoding. We directly predict the other parameters of 3D Gaussians at each pixel with a light-weight decoder ϕ_{GS} as:

$$\begin{aligned} \mathbf{G}^i = [\mathbf{q}^i, \boldsymbol{s}^i, \alpha^i, \mathbf{c}^i] = \phi_{GS}(\mathbf{F}_{gs}), \\ \mathbf{F}^i_{GS} = \mathbf{F}^i_{2D} \oplus \mathbf{X}^i \oplus \mathbf{F}^i_{3D}, \quad \mathbf{F}_{2D} = \phi_{2D}(\mathbf{I}^i), \end{aligned} \tag{5}$$

where $i = \{t-1, t\}$, $\oplus$ denotes the channel-wise concatenation, ϕ_{2D} denotes the 2D image feature extractor, $\mathbf{q}^i \in \mathbb{R}^{H\times W\times 4}$ and $\mathbf{s}^i \in \mathbb{R}^{H\times W\times 3}$ represent rotation quaternions and scales of pixel-aligned Gaussians at image $\mathbf{I}^i$. We incorporate an additional image feature extractor because the coarse predictor is frozen and cannot be trained by our monocular video setting. Extracting new image features is essential for decoding Gaussians, especially for texture-related properties. Experiments show the importance of the image feature extractor. Then covariance matrix is built with $\mathbf{\Sigma}^i = \mathbf{R}(\mathbf{q}^i)\mathbf{s}\mathbf{s}^{\mathrm{T}}\mathbf{R}(\mathbf{q}^i)^{\mathrm{T}}$. Note that $\bar{\mathbf{G}}$ is not the final Gaussians since it needs to be merged into the previous Gaussian set following Sec. 3.4.

3.4. Feed-Forward ADC

With pixel-aligned Gaussian parameters $\mathbf{G}^t$ of T images, previous methods [2, 3, 24, 25] always naively take the union of Gaussians in all images as the final prediction, i.e., $\mathcal{G} = \bigcup_{t=1}^{T} \mathbf{G}^i$ and $|\mathcal{G}| = T \times H \times W$. However, this approach is both memory-intensive and inefficient in rendering, particularly when dealing with the continuous input of video frames during online reconstruction. Our key observation is that the matched Gaussian pairs in neighboring frames are excessive and prunable since they consistently share similar attributes in shape and color, and are closely distributed, which is validated in Tab. 1. It is noted that we have already acquired dense pixel-wise matches between neighboring frames from Eq. (4). Therefore, we propose a novel feed-forward Adaptive Density Control strategy based on revisiting these dense correspondences.

Feature aggregation. The primary advantage of pixel-based correspondences is that they are able to convert the computationally intensive 3D Gaussian aggregation process into a more efficient 2D pixel-wise one. This significantly enhances computational efficiency. Initially, the feature of Gaussian parameters $\mathbf{F}^t_{gs}$ of the frame $\mathbf{I}^t$ can be aligned to the previous frame $\mathbf{I}^{t-1}$ using the following wrapping:

$$\mathbf{F}^{t|t-1}_{GS}(j) = \begin{cases} \mathbf{F}^t_{GS}(k) & \text{if } (j,k) \in \overline{\mathcal{M}}^{t-1,t} \\ \mathbf{F}^{t-1}_{GS}(j) & \text{else,} \end{cases} \tag{6}$$

where $\mathbf{F}^{t|t-1}_{GS}$ denotes the feature of Gaussian primitives in the frame $\mathbf{I}^t$ aligned to $\mathbf{I}^{t-1}$. Instead of using the raw correspondences set $\mathcal{M}^{t-1,t}$ in Eq. (4), we use the extended set $\overline{\mathcal{M}}^{t-1,t}$, which includes matches of neighboring pixels such as $(i+1, j+1)$ in addition to the initial (i, j). It serves as a type of anti-aliasing technique to reduce void pixels within the wrapped feature map. As for the unmatched pixels, we simply replicate the corresponding feature vector within $\mathbf{F}^{t-1}_{gs}$. With the aligned feature, the Gaussian feature of two frames can be merged by modifying Eq. (5), taking the form:

$$\hat{\mathbf{G}}^{t|t-1} = [\hat{\mathbf{q}}^t, \hat{\boldsymbol{s}}^t, \hat{\alpha}^t, \hat{\mathbf{c}}^t] = \phi_{MG}(\mathbf{F}^{t|t-1}_{GS} \oplus \mathbf{F}^{t-1}_{GS}), \tag{7}$$

where ϕ_{MG} denotes the MergeNet that simply consists of two convolutional layers, which merges features and decodes them to Gaussian primitives. In this way, every matched pair of Gaussians between $\mathbf{I}^t$ and $\mathbf{I}^{t-1}$ is aggregated into a single one, highly reducing the number of Gaussians. The final aggregated Gaussian set is $\mathcal{G}^t = \mathcal{G}^{t-1} \cup \mathbf{G}^{t|t-1}$.

Without gradient back-propagation of rendering loss in the original 3DGS paper [12], our ADC process runs exceptional fast. In terms of the input group, the final prediction of Gaussian primitives $\mathcal{G}^t$ consists of the merged Gaussians and the unmatched Gaussians at each frame.

3.5. Loss Functions

The optimization of the 2D feature extractor ϕ_{2D}, the Gaussian decoder network, ϕ_{GS} involves both a rendering loss function and a reconstruction loss function:

$$\begin{aligned} \mathcal{L}(\mathbf{I}^i, \hat{\mathbf{I}}^i, \hat{\mathbf{I}}^i_M) &= \mathcal{L}_{\text{render}}(\mathbf{I}^i, \hat{\mathbf{I}}^i) + \mathcal{L}_{\text{recon}}(\hat{\mathbf{I}}^i, \hat{\mathbf{I}}^i_M) \\ &= \| \mathbf{I}^i - \hat{\mathbf{I}}^i \|_2 + \lambda \| \mathbf{I}^i - \hat{\mathbf{I}}^i \|_{\text{LPIPS}} + \| \hat{\mathbf{I}}^i_{\text{M}} - \hat{\mathbf{I}}^i \|_2, \end{aligned} \tag{8}$$

where $\mathbf{I}^i$ is the ground truth frame, $\hat{\mathbf{I}}^i$ is the rendered image with full Gaussian primitives before the merge process, and $\hat{\mathbf{I}}^i_{\text{M}}$ is the rendered image with merged Gaussians. Due to the lightweight nature of the two networks, the algorithm converges quickly within thousands of steps. The reconstruction loss term facilitates the merge network in fusing the pixel-aligned Gaussian parameters from different frames, aiming to render the same frame as before the merge process but with significantly fewer Gaussian primitives.

4. Experiments

StreamGS operates on an image stream of a scene, jointly predicting the corresponding poses and Gaussians in a feed-forward manner. To assess its performance, we evaluate our

	mean ↓	Opa. ↓	Rot. ↓	Scale ↓	SH ↓
Random Pick	0.41	0.10	3.31	2.47	0.27
Matched Pairs	**0.18**	**0.04**	**3.06**	**1.28**	**0.04**

Table 1. Similarities of attributes between matched GS across adjacent frames on RE10K. Random Pick refers to **randomly** picking two GS from two frames separately, while Matched Pairs refers to our **matched** GS defined in Eq. (4).

method on the task of novel view synthesis from monocular videos, as detailed in Sec. 4.1. Additionally, we validate the effectiveness of the proposed alignment module and the efficiency of the Gaussians merge process, as described in Sec. 4.2.

Baselines and datasets. To the best of our knowledge, StreamGS is the first method to reconstruct unposed videos in a feed-forward manner. Consequently, we compare our method separately with pose-free 3DGS works and generalizable splatting methods, including pixelSplat [2], MVSplat [3], and CF-3DGS [9]. PixelSplat and MVSplat are representative methods of generalizable Gaussian Splatting, but they both rely on known camera poses and intrinsics. In contrast, CF-3DGS is a pose-free method but requires optimization loops to align poses and Gaussians. For a comprehensive comparison, we evaluate the methods on large-scale datasets with diverse scenes, including RE10K [38], ACID [17], ScanNet [5], DL3DV [16], and MVImgNet [33]. Each dataset comprises monocular video sequences with per-frame camera pose annotations.

Implementation details. Both the 2D image feature extractor ϕ_{2D} and the MergeNet ϕ_{GS} are double-layer convolutional networks. For fairness, StreamGS and other generalizable methods are trained on the identical training split of the RE10K dataset using the Adam [13] optimizer with the same learning rate of 2×10^{-4} and a cosine scheduler. The parameter λ in Eq. 8 is set to 0.05. All methods are trained for 30K iterations and tested on a single NVIDIA Tesla A100 80GB GPU. The images are resized to 224×224 and the batch size is set to 14. For the non-generalizable method CF-3DGS, we follow its original setting [9]. Note that CF-3DGS is evaluated only on a subset of the full test set due to its low reconstruction efficiency, as shown in Fig. 5. Unlike pose-dependent methods, CF-3DGS and our method require poses of novel views for rendering. CF-3DGS freezes the trained Gaussian model and performs additional optimization steps to learn the poses. Our method, as described in Sec. 3.3, carries out pose alignment and refinement processes to estimate the poses. More details can be referred to supplementary materials.

4.1. Novel View Synthesis

4.1.1. Reconstruction Quality

Quantitative comparison. We compare StreamGS with baseline methods on the quality and efficiency of novel view synthesis. Following previous 3D-GS research [12], we report PSNR, SSIM [28], and LPIPS [37] as metrics of reconstruction quality. The quantitative results are shown in Tab. 2. On the source domain RE10K [38], existing state-of-the-art generalizable methods demonstrate competitive scores, with MVSplat even outperforming the optimization-based CF-3DGS. However, their performance degrades significantly on out-of-domain datasets such as DL3DV [16] and MVImgNet [33], as these datasets contain various scene types, including outdoor environments and more complex indoor scenes with different objects and illumination conditions compared to RE10K. Naturally, CF-3DGS, which is based on thousands of optimization steps, performs well on the aforementioned datasets. However, its PSNR slightly decreases on ScanNet due to more irregular camera movements and increased motion blur in the frames, posing a challenge for CF-3DGS in recovering camera poses. PixelSplat performs well on ScanNet, thanks to its robust feature extraction backbone trained on ImageNet [6], while MVSplat achieves the lowest score. According to the table, StreamGS consistently achieves scores comparable to all other methods. Its PSNR on MVImgNet and DL3DV is significantly higher than that of PixelSplat and MVSplat, even without given poses and intrinsics, demonstrating our method's superior generalizability on unseen datasets with significant domain gaps.

Qualitative comparison. A qualitative comparison with state-of-the-art methods is also presented in Fig. 3 and Fig. 4. While all methods demonstrate high-quality novel view rendering on RE10K [38], our method exhibits superior robustness on out-of-domain datasets. Due to the significant domain gap including texture, illumination and camera motion, both MVSplat and pixelSplat fail to predict accurate depths for the printer shown in the second row, resulting in severe floating artifacts in the rendered image. The third row shows the reconstruction of a plaza from DL3DV [16]. Similarly, since outdoor scenes are much less represented in RE10K [38], MVSplat struggles to extract stereo cues from text-less skies and objects with disparate illumination, deteriorating the rendering quality. The final row shows a bench scene in MVImgNet, on which baseline generalizable methods also fails due to the similar domain gap issues. These cases also demonstrate that the generalizability of pixelSplat surpasses that of MVSplat. As the figure shows, CF-3DGS also does not perform well on some outdoor scenes. In contrast, the visual quality of our method on out-of-domain datasets remains high.

4.1.2. Reconstruction Efficiency

In addition to rendering quality, we also compare the reconstruction efficiency of our method with baseline models. Fig. 5 illustrates a plot of reconstruction quality, measured by the average PSNR reported in Tab. 2, versus ef-

	PF	G	RE10K [38] (Source Domain)			DL3DV [16]			MVImgNet [33]			ScanNet [5]			ACID [17]		
			PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
pixelSplat [2]	✘	✔	22.64	0.21	0.75	19.43	0.39	0.50	19.22	0.49	0.60	27.56	0.19	0.82	**29.93**	**0.14**	**0.84**
MVSplat [3]	✘	✔	**23.54**	**0.13**	**0.90**	17.84	0.36	0.45	16.29	0.50	0.53	26.25	0.19	0.82	28.83	0.14	0.86
CF-GS [9]	✔	✘	23.46	0.24	0.75	19.93	0.31	0.62	**26.33**	**0.33**	**0.88**	22.88	0.42	0.76	28.16	0.19	0.81
StreamGS	✔	✔	22.42	0.17	0.83	**20.54**	**0.24**	**0.64**	25.05	0.31	0.79	**28.43**	**0.16**	**0.86**	28.50	0.15	0.84

Table 2. Quantitative comparison with existing state-of-art methods on Novel View Synthesis of monocular videos. **PF** indicates whether the method is **pose-free**. **G** indicates whether the methods is generalizable. Our method consistently achieves scores comparable to state-of-the-art methods that are either not pose-free or lack generalizability. In the table, the best result is highlighted in **bold**.

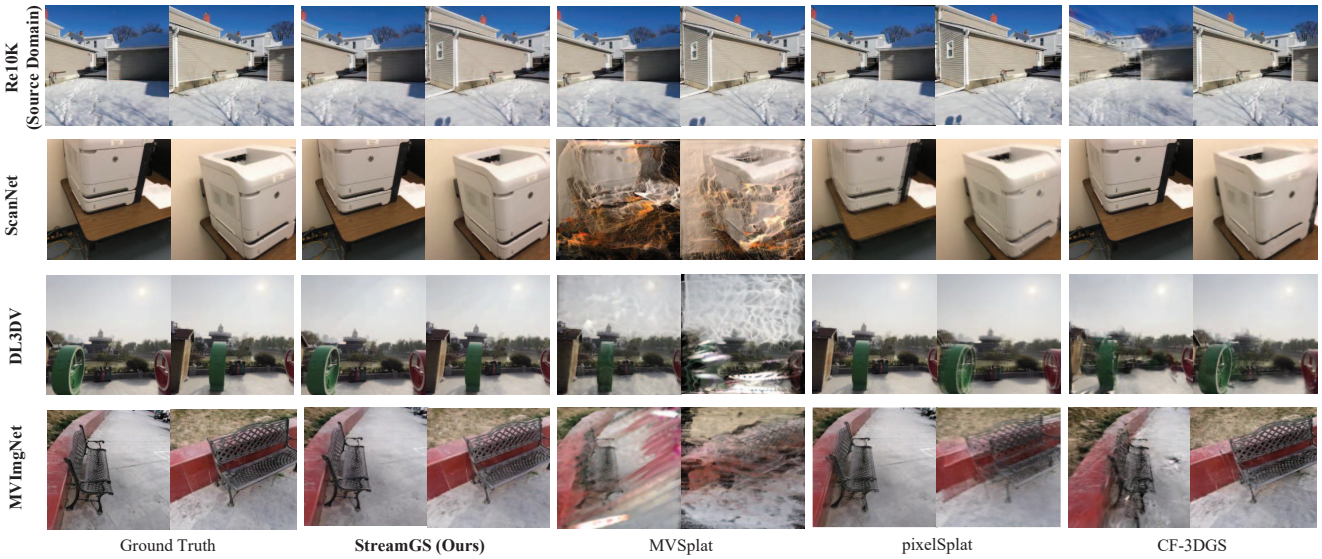

Figure 3. Qualitative comparison on novel view synthesis. We show the results on both source domain, RE10K [38], and other domains, ScanNet [5], DL3DV [16] and MVImgNet [33]. All generalizable methods are trained only on RE10K and tested on the other datasets. StreamGS outperforms other methods in several challenging scenarios, especially for the out-of-domain data.

	Coarse Pred (Sec. 3.2)		Ada. Refine (Sec. 3.3)		ADC (Sec. 3.4)		Total	
	time/s	param/M	time/s	param/M	time/s	param/M	time/s	param/M
StreamGS	0.02	656.74	0.08	1.83	0.01	0.04	0.11	658.61

Table 3. Efficiency metrics of each component.

ficiency, measured by the processing time per frame (FPS). The figure shows that StreamGS achieves second place with a PSNR of 23.1, only 0.05 lower than CF-3DGS [9]. This indicates that our method achieves nearly the same rendering quality as the best model. However, thanks to the feed-forward design, StreamGS is **150 times** faster than CF-3DGS, predicting Gaussian primitives for up to 9 frames within one second. Without known camera information, our method involves additional alignment and pose estimation processes across frames, which limits the inference speed, making it slower than MVSplat [3] and pixelSplat [2]. However, according to the scores reported in Tab. 2 and Fig. 5, our method is more generalizable and achieves better rendering quality than these methods. Tab. 3 shows the efficiency of each component of our method.

4.2. Ablation Study

In this section, we discuss the effectiveness of our main design about joint refinement (in Sec. 3.3) and feed-forward ADC module (in Sec 3.4). More ablation studies on framework design can be found in supplementary materials.

4.2.1. Effectiveness of Joint Refinement

	RE10K [38]			ACID [17]		
	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
w/o refine.	17.25	0.32	0.58	17.23	0.39	0.48
w/ refine	**22.42**	**0.17**	**0.83**	**28.50**	**0.15**	**0.84**

Table 4. Evaluation of effectiveness of joint refinement.

Joint refinement of cameras and centers of Gaussians plays a crucial role in the success of our method. To validate the effectiveness of joint refinement, we conduct an ablation study by skipping the refinement process during inference. In other words, the poses and intrinsics are directly estimated by Eq. 2 and 3. Tab. 4 shows the quantitative comparison between the two settings. Without joint refinement, Gaussian primitives are cast from shifted origins of the camera with erroneous orientations, and the poses of novel views are also inaccurate, causing the rendered images to be shifted and distorted. This severely deteriorates the rendering quality, as shown in Tab. 4. The PSNR

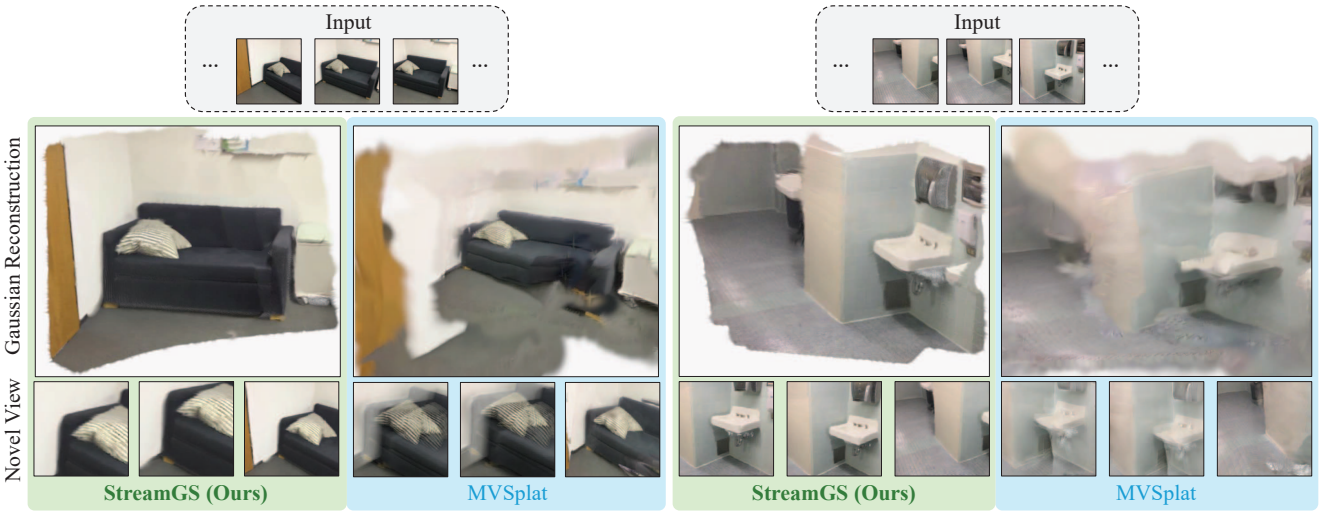

Figure 4. Visual comparison of Gaussian reconstruction and novel view synthesis from image streams with ScanNet [5] dataset. Unlike MVSplat [3], which struggles with view aggregation, our results show significantly better visual quality on OOD data. Note that Our StreamGS and MVSplat are both trained with RE10K [38] data, and MVSplat needs predetermined cameras.

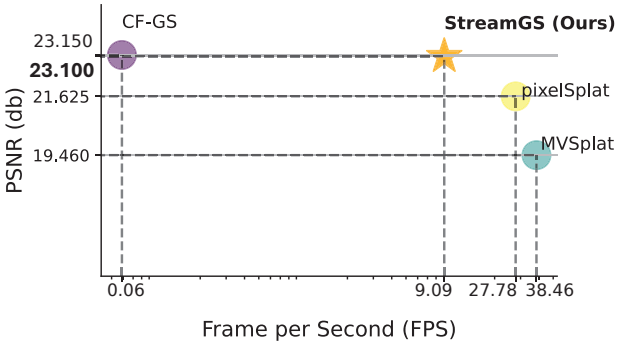

Figure 5. Reconstruction speed measured by frames processed per second (FPS). The x-axis is log-scaled for the better visualization.

of the images rendered with direct estimation decreases by 23.06% and 38.62% on the RE10K [38] and ACID [17] datasets, respectively.

4.2.2. Effectiveness of Gaussian Merging Process

	MVImgNet [33]		ACID [17]	
	Compress. Ratio ↑	PSNR ↑	Compress. Ratio ↑	PSNR ↑
w/o merge.	1.00	25.70	1.00	29.21
merge all	**1.58**	25.05	**1.68**	28.07

Table 5. Evaluation of the efficiency improvements of the Gaussian merging process and its impact on rendering quality.

During the feed-forward ADC, StreamGS prunes pixel-aligned Gaussians through a merging process. To evaluate the memory efficiency improvement and its impact on rendering quality, we compare the average number of Gaussians per frame and PSNR before and after the merging process. We define the compression ratio of Gaussians during the merging process as the ratio of the average number of Gaussians per frame, i.e., $H \times W$, to that after the merging process. The metrics are reported in Tab. 5. The results demonstrate that the merging process can prune Gaussians per frame by 36.71% and 40.48% on MVImgNet [33] and ACID [17], respectively. Meanwhile, the PSNR scores after the merging process only slightly decrease by 2.53% and 3.90%, respectively. This ablation study demonstrates that the designed Gaussian merging process efficiently reduces memory usage during reconstruction and rendering, with a negligible impact on reconstruction quality.

5. Conclusion

We propose a novel and holistic generalizable pose-free reconstruction pipeline named *StreamGS*, dedicated to the online reconstruction of endless unposed image streams, such as monocular videos. To the best of our knowledge, our method is the first generalizable model capable of predicting Gaussians corresponding to the input stream in a feed-forward manner, without relying on known poses and intrinsics. Compared to pose-free but optimization-based methods, our method achieves comparable reconstruction quality while reducing the learning time to within several milliseconds, avoiding optimization steps. Compared to other generalizable methods, StreamGS eliminates the dependence on poses and intrinsics and manages to reconstruct more accurate scenes on out-of-domain datasets, demonstrating better domain generalizability.

Limitations. Although our method runs fast, joint refinement process still includes additional time costs, making it slower than MVSplat [3]. Moreover, our approach encounters common reconstruction challenges, including textureless regions and long sequences.

References

[1] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4160–4169, 2023. 2, 3

[2] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. 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*, pages 19457–19467, 2024. 2, 3, 5, 6, 7

[3] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. *arXiv preprint arXiv:2403.14627*, 2024. 2, 3, 5, 6, 7, 8

[4] Zezhou Cheng, Carlos Esteves, Varun Jampani, Abhishek Kar, Subhransu Maji, and Ameesh Makadia. Lu-nerf: Scene and pose estimation by synchronizing local unposed nerfs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18312–18321, 2023. 3

[5] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 6, 7, 8

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[7] Zhiwen Fan, Wenyan Cong, Kairun Wen, Kevin Wang, Jian Zhang, Xinghao Ding, Danfei Xu, Boris Ivanovic, Marco Pavone, Georgios Pavlakos, et al. Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds. *arXiv preprint arXiv:2403.20309*, 2024. 2

[8] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 5

[9] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A. Efros, and Xiaolong Wang. Colmap-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20796–20805, 2024. 2, 3, 6, 7

[10] Yiming Gao, Yan-Pei Cao, and Ying Shan. Surfelnerf: Neural surfel radiance fields for online photorealistic reconstruction of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 108–118, 2023. 1, 3

[11] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 2

[12] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1, 3, 5, 6

[13] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Ep n p: An accurate o (n) solution to the p n p problem. *International journal of computer vision*, 81:155–166, 2009. 5

[15] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r, 2024. 2

[16] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, 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*, pages 22160–22169, 2024. 6, 7

[17] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *ICCV*, 2021. 6, 7, 8

[18] Tianqi Liu, Guangcong Wang, Shoukang Hu, Liao Shen, Xinyi Ye, Yuhang Zang, Zhiguo Cao, Wei Li, and Ziwei Liu. Mvsgaussian: Fast generalizable gaussian splatting reconstruction from multi-view stereo. In *European Conference on Computer Vision*, pages 37–53. Springer, 2025. 3

[19] Andreas Meuleman, Yu-Lun Liu, Chen Gao, Jia-Bin Huang, Changil Kim, Min H Kim, and Johannes Kopf. Progressively optimized local radiance fields for robust view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16539–16548, 2023. 3

[20] Jongmin Park, Minh-Quan Viet Bui, Juan Luis Gonzalez Bello, Jaeho Moon, Jihyong Oh, and Munchurl Kim. Splinegs: Robust motion-adaptive spline for real-time dynamic 3d gaussians from monocular video. *arXiv preprint arXiv:2412.09982*, 2024. 2

[21] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1, 3

[22] Qiuhong Shen, Zike Wu, Xuanyu Yi, Pan Zhou, Hanwang Zhang, Shuicheng Yan, and Xinchao Wang. Gamba: Marry gaussian splatting with mamba for single view 3d reconstruction. *arXiv preprint arXiv:2403.18795*, 2024. 3

[23] Liang Song, Guangming Wang, Jiuming Liu, Zhenyang Fu, Yanzi Miao, et al. Sc-nerf: Self-correcting neural radiance field with sparse views. *arXiv preprint arXiv:2309.05028*, 2023. 2

[24] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10208–10217, 2024. 2, 5

[25] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 3, 5

[26] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 2

[27] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 1, 2, 4

[28] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

[29] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv preprint arXiv:2403.14621*, 2024. 3

[30] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19595–19604, 2024. 1, 2, 3

[31] Xuanyu Yi, Zike Wu, Qiuhong Shen, Qingshan Xu, Pan Zhou, Joo-Hwee Lim, Shuicheng Yan, Xinchao Wang, and Hanwang Zhang. Mvgamba: Unify 3d content generation as state space sequence modeling. *arXiv preprint arXiv:2406.06367*, 2024. 3

[32] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4578–4587, 2021. 3

[33] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Tianyou Liang, Guanying Chen, Shuguang Cui, and Xiaoguang Han. Mvimgnet: A large-scale dataset of multi-view images. In *CVPR*, 2023. 6, 7, 8

[34] Vladimir Yugay, Yue Li, Theo Gevers, and Martin R Oswald. Gaussian-slam: Photo-realistic dense slam with gaussian splatting. *arXiv preprint arXiv:2312.10070*, 2023. 2, 3

[35] Haojun Zhang, Yuan Yao, and Xuefeng Yan. Improved end-to-end multilevel nerf-based dense rgb-d slam. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pages 132–146. Springer, 2024. 2, 3

[36] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19. Springer, 2025. 3

[37] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[38] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ACM Trans. Graph.*, 37, 2018. 6, 7, 8

[39] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12786–12796, 2022. 3

[40] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. In *2024 International Conference on 3D Vision (3DV)*, pages 42–52. IEEE, 2024. 2, 3

[41] Zi-Xin Zou, Shi-Sheng Huang, Yan-Pei Cao, Tai-Jiang Mu, Ying Shan, and Hongbo Fu. Mononeuralfusion: Online monocular neural 3d reconstruction with geometric priors. *arXiv preprint arXiv:2209.15153*, 2022. 2, 3

[42] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. 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*, pages 10324–10335, 2024. 3