

No Pose at All: Self-Supervised Pose-Free 3D Gaussian Splatting from Sparse Views

Ranran Huang, Krystian Mikolajczyk
Imperial College London

{r.huang24, k.mikolajczyk}@imperial.ac.uk

Abstract

We introduce *SPFSplat*, an efficient framework for 3D Gaussian splatting from sparse multi-view images, requiring **no ground-truth poses** during training or inference. It employs a shared feature extraction backbone, enabling simultaneous prediction of 3D Gaussian primitives and camera poses in a canonical space from unposed inputs within a single feed-forward step. Alongside the rendering loss based on estimated novel-view poses, a reprojection loss is integrated to enforce the learning of pixel-aligned Gaussian primitives for enhanced geometric constraints. This pose-free training paradigm and efficient one-step feed-forward design make *SPFSplat* well-suited for practical applications. Remarkably, despite the absence of pose supervision, *SPFSplat* achieves state-of-the-art performance in novel view synthesis even under significant viewpoint changes and limited image overlap. It also surpasses recent methods trained with geometry priors in relative pose estimation. Code and trained models are available on our project page: <https://ranruang.github.io/spfsplat/>.

1. Introduction

Recent advancements in 3D reconstruction and novel view synthesis (NVS) have been driven by Neural Radiance Fields (NeRFs) [29] and 3D Gaussian splatting (3DGS) [22]. A standard NVS training pipeline [4, 5, 8, 37, 47, 49] reconstructs a 3D scene from input views and optimizes it by aligning rendered novel views with ground-truth images.

State-of-the-art (SOTA) methods based on NeRF [5, 47] and 3DGS [4, 8, 40, 48, 51] typically employ geometry-aware architectures, relying on camera poses estimated using Structure-from-Motion (SfM) [36] to reconstruct 3D scenes, as illustrated in Fig. 1 (a). However, the acquisition of camera poses from SfM is computationally expensive and often unreliable in sparse-view scenarios due to insufficient correspondences, limiting the applicability of these *pose-required* methods. To address this, recent research has focused on novel view synthesis under pose-free settings.

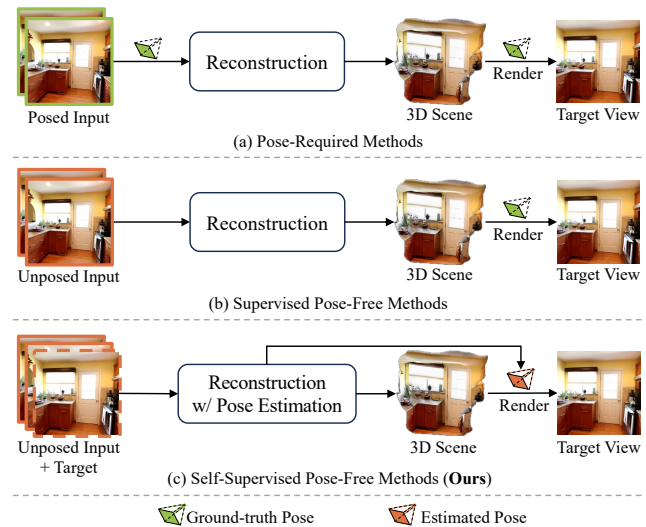


Figure 1. Comparison of three **training** pipelines for sparse-view 3D scene reconstruction in novel view synthesis. For simplicity, the image rendering loss on the rendered target view is omitted. Our self-supervised pose-free pipeline estimates target-view poses to optimize 3D scene representations reconstructed from unposed images, thereby eliminating the reliance on ground-truth poses during training.

Existing pose-free methods reconstruct 3D scenes from unposed images by learning in a canonical space [20, 37, 44, 49], leveraging latent scene representations [30, 33], or jointly optimizing both input-view camera poses and 3D scene representations [7, 17, 25]. Although these methods do not require known input-view poses at inference, they are still trained using rendering losses given ground-truth poses at novel viewpoints, as shown in Fig. 1 (b). We therefore categorize these approaches as *supervised pose-free* methods. As a result, their training remains confined to datasets with known camera poses, limiting scalability to large-scale real-world data without pose annotations.

This raises a critical question: **Are ground-truth novel-view poses truly indispensable for optimizing 3D scenes during training?** One solution is to optimize the 3D scenes using poses estimated from the model, referred to as the

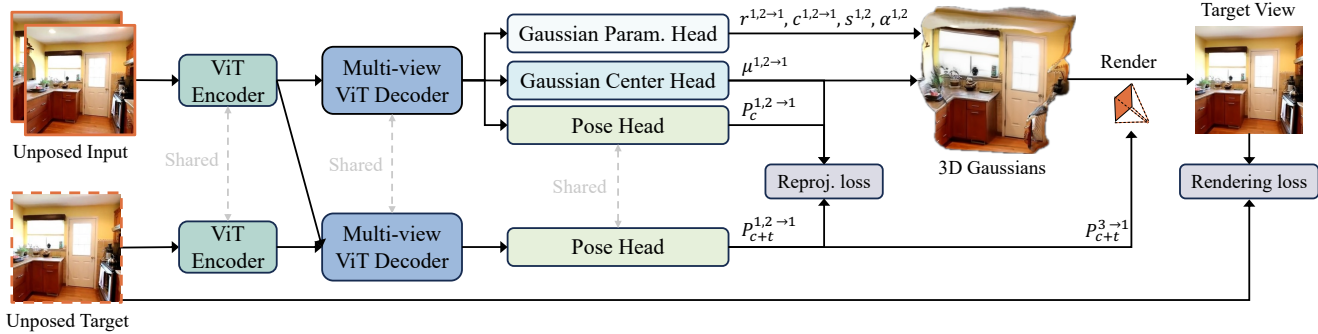


Figure 2. Training pipeline of SPFSplat. Three specialized heads are integrated into a shared ViT backbone, simultaneously predicting Gaussian centers, additional Gaussian parameters, and camera poses from unposed images in a canonical space, where the first input view serves as the reference. Only the context-only branch (above) is used during inference, while the context-with-target branch (below) is employed exclusively during training to estimate target poses, which are used for rendering loss supervision. Additionally, a reprojection loss enforces alignment between Gaussian centers and their corresponding pixels, using the estimated context poses from both branches. Our method jointly optimizes 3D Gaussians and poses, improving geometric consistency and reconstruction quality.

self-supervised pose-free paradigm in Fig. 1 (c). However, this presents an inherent challenge: since the rendering loss intrinsically couples the learning of 3D scene geometry and camera poses, pose errors can degrade reconstruction quality, which further hampers pose estimation. Such mutual dependency creates a feedback loop that can potentially lead to unstable training or even divergence. Recent self-supervised pose-free approaches [16, 21] struggle to mitigate this issue primarily due to their use of separate modules for scene reconstruction and pose estimation, discouraging the learning of consistent feature representations across the two tasks and impairing geometric alignment. Consequently, these methods still lag far behind SOTA pose-required and supervised pose-free methods [4, 8, 49].

To address the challenge, we introduce SPFSplat, a self-supervised **pose-free** approach for 3D Gaussian splatting from sparse views, where the novel-view poses are estimated given target images during training. As shown in Fig. 2, SPFSplat employs a shared backbone for feature extraction, equipped with dedicated heads for predicting 3D Gaussian primitives and camera poses. This unified architecture not only enhances computational efficiency but also facilitates joint feature learning for scene reconstruction and pose estimation, improving geometric consistency and mitigating the destabilizing feedback loop. This is achieved by enabling 3D geometry to benefit from accurate, context-aware camera alignment and allowing pose predictions to leverage global scene context, as a form of mutual reinforcement. Furthermore, we draw inspiration from canonical-space-based methods [20, 23, 37, 45, 49] and directly predict 3D Gaussian primitives relative to the reference view to reduce the impact of pose errors on scene geometry. We also complement the rendering loss with a reprojection loss that explicitly aligns the predicted Gaussians with their corresponding image pixels, imposing stronger geometric constraints to enhance training stability.

We make the following key contributions: (1) We propose SPFSplat, an efficient framework that simultaneously predicts 3D Gaussians and camera poses in a canonical space from sparse views without requiring any ground-truth pose annotations during training or inference. It incorporates a Gaussian prediction module and a lightweight pose head, enabling joint optimization of scene reconstruction and pose estimation using rendering and reprojection losses. (2) To the best of our knowledge, SPFSplat is the first self-supervised pose-free method that outperforms SOTA pose-required and supervised pose-free NVS approaches even under extreme viewpoint changes and limited image overlap. (3) Despite the absence of pose supervision, our feed-forward relative pose estimation is not only highly efficient but also outperforms recent SOTA methods relying on the supervision of geometric priors.

2. Related Work

2.1. Novel View Synthesis

NeRF [29] and 3DGS [22] have shown impressive results in photorealistic NVS. While early methods require dense input views for high-quality output, recent approaches [4, 5, 8, 37, 40, 47–49, 51] focus on 3D reconstruction and novel view synthesis from sparse-view images. The typical NVS pipeline involves reconstructing 3D scenes from input views, followed by optimization with synthesized images aligned to ground-truth targets. Based on their reliance on ground-truth poses during training and inference, existing methods can be categorized into three groups: pose-required, supervised pose-free, and self-supervised pose-free methods, as shown in Fig. 1.

Pose-Required Methods rely on accurate camera poses during both training and inference to reconstruct 3D scenes using various geometric operations [4, 5, 8, 40, 47, 48, 51]. These include constructing cost volumes for multi-view ag-

gregation [5, 47], leveraging epipolar transformers or cost volumes based on feature matching for depth estimation and Gaussian primitive prediction [4, 8], and encoding camera poses via Plücker ray embeddings [40, 48, 51]. While effective, such methods rely on Structure-from-Motion (SfM), which is computationally expensive and often unreliable in sparse-view scenarios. Although recent pose estimation methods [23, 24, 42, 45, 50] attempt to mitigate these limitations, they still struggle with low-overlap or texture-less data. Consequently, these pose-required approaches are not applicable in unposed settings.

Supervised Pose-Free Methods enable 3D reconstruction from unposed images, relaxing the requirement for camera poses at inference time. For instance, methods such as [30, 33] encode unposed images into latent scene representations, while [7, 17, 25] jointly optimize camera poses and NeRF representations. Approaches like LEAP [20] and PF-LRM [44] define neural volumes in the canonical view’s local camera coordinate system. Similarly, Splatt3R [37] and NoPoSplat [49] predict 3D Gaussians in a canonical space. However, these methods still rely on ground-truth camera poses during training through image rendering losses [20, 30, 33, 37, 44, 49], pose prediction loss [17], or coarse initialization [25, 41]. Therefore, their training remains limited to datasets with ground-truth poses.

Self-Supervised Pose-Free Methods completely eliminate the reliance on ground-truth poses during both training and inference. For instance, Nope-NeRF [1], CF-3DGS [14], and FlowCam [38] reconstruct 3D scenes and estimate camera poses incrementally by re-rendering dense video sequences. However, they are limited to continuous video frames and do not generalize well to sparse views. Recent self-supervised pose-free methods, such as PF3plat [16] and SelfSplat [21] attempt to estimate camera poses from sparse views. Specifically, PF3plat relies on off-the-shelf feature descriptors (LightGlue [26]) and RANSAC-based pose initialization, resulting in a pipeline that is neither efficient nor end-to-end trainable. In contrast, SelfSplat employs cross-view U-Nets [32] to predict depth and pose separately. Despite their differences, both methods adopt separate modules for pose estimation and scene reconstruction, leading to unshared and inconsistent features, poor geometric alignment, and increased computational overhead. Inaccurate poses further directly degrade reconstruction by corrupting the lifted 3D points, amplifying reconstruction errors and exacerbating the feedback loop instability.

Our method also adopts a self-supervised, pose-free paradigm, requiring no ground-truth poses during training or inference. In contrast to [16, 21], we jointly optimize 3D Gaussians and camera poses via a shared backbone in a canonical space [23, 37, 45, 49], guided by both image rendering and reprojection losses. This unified design ensures that pose estimation is informed by the same scene

geometry that drives Gaussian prediction, encouraging geometric alignment between the scene representation and the predicted poses and enhancing training stability.

2.2. Structure-from-Motion (SfM)

Structure-from-Motion (SfM) is a fundamental problem in computer vision [15], involving the simultaneous reconstruction of sparse 3D maps and estimation of camera parameters from a set of images. Recent advances have integrated learning-based approaches into various SfM components. Enhancements include more robust feature descriptors [9, 11, 18], improved image matching [35], detector-free matching [39], and neural bundle adjustment [25]. Moreover, fully differentiable SfM pipelines have been introduced [43, 45]. Unlike VGGsSfM [43], which focuses on end-to-end sparse reconstruction, DUST3R [45] enables dense 3D reconstruction without requiring camera parameters and has been successfully applied to pose estimation, monocular depth estimation and 3D reconstruction. As an extension, MAST3R [23] integrates feature matching with DUST3R and improves local feature representation.

Similar to SfM methods, our approach jointly predicts 3D points and camera poses. The image rendering and reprojection losses used during training can be interpreted as a form of bundle adjustment, further jointly refining and aligning the estimated scene representation and poses.

3. Method

3.1. Problem Formulation

We aim to learn a feed-forward network that reconstructs 3D Gaussians from N unposed input images $\{\mathbf{I}^v\}_{v=1}^N$ while simultaneously estimating the camera poses. During training, the 3D Gaussians are optimized by synthesizing photo-realistic images $\hat{\mathbf{I}}^t$ from the estimated poses at target view t , thereby eliminating the need for ground-truth poses.

3D Gaussian Reconstruction. Following [37, 49], we predict 3D Gaussians in a canonical 3D space where the first input view \mathbf{I}^1 serves as the global reference coordinate frame. The network is formulated as:

$$f_{\theta} : \{\mathbf{I}^v\}_{v=1}^N \mapsto \{\mathcal{G}^{v \rightarrow 1}\}_{v=1, \dots, N}, \quad (1)$$

where $\mathcal{G}^{v \rightarrow 1} = \{(\boldsymbol{\mu}_j^{v \rightarrow 1}, \mathbf{r}_j^{v \rightarrow 1}, \mathbf{c}_j^{v \rightarrow 1}, \alpha_j^v, \mathbf{s}_j^v)\}_{j=1, \dots, H \times W}$ represents the pixel-aligned Gaussians for \mathbf{I}^v , represented in the coordinate frame of \mathbf{I}^1 . Each Gaussian is parameterized by center $\boldsymbol{\mu} \in \mathbb{R}^3$, rotation quaternion $\mathbf{r} \in \mathbb{R}^4$, scale $\mathbf{s} \in \mathbb{R}^3$, opacity $\alpha \in \mathbb{R}$, and spherical harmonics (SH) $\mathbf{c} \in \mathbb{R}^k$ with k degrees of freedom.

Pose Estimation. We introduce a pose network f_{ϕ} to estimate the relative transformation from the view \mathbf{I}^v to the reference view \mathbf{I}^1 . The estimated relative transformation from \mathbf{I}^v to \mathbf{I}^1 is denoted as $\mathbf{P}^{v \rightarrow 1} = [\mathbf{R}^{v \rightarrow 1} | \mathbf{T}^{v \rightarrow 1}]$, where $\mathbf{R}^{v \rightarrow 1} \in \mathbb{R}^{3 \times 3}$ represents the rotation matrix, and

$\mathbf{T}^{v \rightarrow 1} \in \mathbb{R}^{3 \times 1}$ represents the translation vector. The pose estimation is formulated as:

$$\mathbf{P}^{v \rightarrow 1} = f_\phi(\mathbf{I}^v, \mathbf{I}^1). \quad (2)$$

Novel View Synthesis. During training, to eliminate the reliance on ground-truth poses for image synthesis at target viewpoints, we also estimate the relative pose from \mathbf{I}^t to \mathbf{I}^1 , which is denoted as $\mathbf{P}^{t \rightarrow 1} = [\mathbf{R}^{t \rightarrow 1} | \mathbf{T}^{t \rightarrow 1}]$. Using the estimated transformation, we render images from novel viewpoints following Eq. 3:

$$\hat{\mathbf{I}}^t = \mathcal{R}(\mathbf{P}^{t \rightarrow 1}, \{\mathcal{G}^{v \rightarrow 1}\}_{v=1, \dots, N}). \quad (3)$$

As in prior work, we assume that intrinsic parameters are available from camera sensor metadata.

3.2. Architecture

As shown in Fig. 2, our method consists of four main components: an encoder, a decoder, a pose head, and Gaussian prediction heads. Both the encoder and decoder are based on Vision Transformer (ViT) architectures [10].

Encoder and Decoder. For each input view, the RGB image is first patchified and flattened into a sequence of image tokens. Following [49], to alleviate scale ambiguity, we encode the intrinsic camera parameters into an additional token using a linear layer, which is then concatenated with the image tokens along the spatial dimension. It is notable that this operation is optional. As demonstrated in Sec. 4.4, our method surpasses existing approaches even without injecting intrinsic parameters into the backbone. Each view’s tokens are first processed individually by a weight-sharing ViT encoder. Next, a ViT decoder equipped with cross-attention aggregates multi-view information. The decoder jointly reasons over token representations across all input views, with each view attending to all other views, facilitating cross-view information exchange to capture spatial relationships and the global 3D scene geometry. In contrast to pairwise architectures [23, 45], this approach supports efficient incorporation of additional input views without significantly increasing the memory or computation cost.

Gaussian Prediction Heads. Following [37, 49], we use two DPT-based heads [31] to infer Gaussian parameters. The first head processes decoder tokens from context views and predicts 3D coordinates for each pixel, defining Gaussian centers. The second head estimates rotation, scale, opacity, and SH coefficients for each Gaussian primitive. As proposed in [4, 8, 49], we incorporate high-resolution skip connections by feeding the original input image into the prediction heads, preserving fine-grained spatial details.

Pose Head. The pose head enables the prediction of poses for input views in a single feed-forward step, and is essential for self-supervised learning of Gaussians. It is built on the same decoder as the Gaussian heads, encouraging

shared geometric knowledge and better alignment between the Gaussians and poses.

Within the pose head, token representations from the encoder and decoder are concatenated, unpatchified, and processed via global average pooling to generate a compact geometric embedding for each view. This embedding is then fed into a lightweight 3-layer MLP, which directly outputs the camera pose as a 10-dimensional representation [6]. The pose is decomposed into translation and rotation for each view. The translation is represented using four homogeneous coordinates [2], while the rotation is encoded in a 6D format, capturing two unnormalized coordinate axes. These axes are normalized and combined via a cross-product operation to construct a full rotation matrix [54]. To compute the relative pose with respect to the reference view, the 10D pose representation is converted into a homogeneous transformation matrix $\mathbf{P}^{v \rightarrow 1} \in \mathbb{R}^{4 \times 4}$. We normalize the camera poses by assigning the first input view the canonical pose $[\mathbf{U} | \mathbf{0}]$, where \mathbf{U} represents the identity matrix, and $\mathbf{0}$ denotes the zero translation vector. More details can be found in the appendix.

During training, to enable image synthesis at target views without ground-truth poses, we introduce an additional input branch that incorporates both context and target views. The encoder tokens from these views are jointly aggregated by the multi-view ViT decoder, after which the target poses are predicted via the pose head. Importantly, Gaussian reconstruction and target pose prediction are decoupled to prevent information leakage. As illustrated in Fig. 2, Gaussian representations are predicted solely from the context views, while target pose estimation leverages information from both context and target views to achieve a more comprehensive understanding of the global geometric structure. This design ensures that the information from the target view does not influence the 3D Gaussian representation, thereby improving generalization to novel viewpoints.

3.3. Loss Function

Image Rendering Loss. Our model is trained using ground-truth target RGB images as supervision. The training loss is formulated as a weighted combination of the L_2 loss and the LPIPS loss [52], formulated as:

$$\mathcal{L}_{\text{render}} = \|\mathbf{I}^t - \hat{\mathbf{I}}^t\|_2 + \gamma \text{LPIPS}(\mathbf{I}^t, \hat{\mathbf{I}}^t), \quad (4)$$

where \mathbf{I}^t and $\hat{\mathbf{I}}^t$ denote the ground-truth and rendered target images, respectively, and γ is a weighting factor that balances perceptual similarity and pixel-level accuracy.

Reprojection Loss. Existing approaches enforce pixel-aligned Gaussian prediction by constraining Gaussian locations along the input viewing rays [4, 8, 16, 21, 48, 51]. On the other hand, canonical-space-based methods [37, 49] rely on ground-truth camera poses to guide the canonical

3D points (Gaussian centers). Both strategies ensure alignment between each pixel and its corresponding 3D point. However, since our model learns 3D Gaussian centers in a canonical space without known camera poses, the network lacks explicit geometric constraints to enforce pixel-aligned Gaussian representation.

A naive solution is to include context views in the rendering loss (Eq. 4) by synthesizing images from them and computing the loss against their ground-truth counterparts. However, this leads to unstable training due to overfitting. Specifically, the network prioritizes improving the rendering quality of the first context view, as the 3D Gaussian space is defined in its camera coordinate, making its rendering independent of the learnable poses. Since the Gaussians from this view already capture sufficient scene information, the model suppresses the contribution of other context views by shifting their Gaussian centers away and adjusting camera poses, ultimately leading to training collapse.

To address this issue, we employ a pixel-wise reprojection loss to jointly optimize 3D points and camera poses [3, 36]. Unlike image-based supervision, reprojection loss enforces explicit geometric constraints, reducing overfitting to context views. Specifically, for each pixel \mathbf{p}_j^v in context view v , we project the corresponding 3D Gaussian center $\boldsymbol{\mu}_j^{v \rightarrow 1}$ from the first camera coordinate frame into 2D pixel coordinates using the estimated pose of view v and minimize the pixel-wise reprojection error. Since context poses can be obtained from both the context-only and context-with-target branches during training (Fig. 2), we enforce consistency by applying reprojection loss to both:

$$\mathcal{L}_{\text{reproj}} = \sum_{v=1}^N \sum_{j=1}^{H \times W} \|\mathbf{p}_j^v - \pi(\mathbf{K}^v, \mathbf{P}^{v \rightarrow 1}, \boldsymbol{\mu}_j^{v \rightarrow 1})\|, \quad (5)$$

where $\mathbf{P}^{v \rightarrow 1} \in \{\mathbf{P}_c^{v \rightarrow 1}, \mathbf{P}_{c+t}^{v \rightarrow 1}\}$, π represents the camera projection function, \mathbf{K}^v denotes the camera intrinsics of view v . $\mathbf{P}_c^{v \rightarrow 1}$ is the relative pose from view v to the canonical frame estimated from the context-only branch, while $\mathbf{P}_{c+t}^{v \rightarrow 1}$ is estimated from both context and target views. By leveraging the reprojection loss, our method enables stable training and efficient optimization of pixel-aligned 3D Gaussians, without requiring ground-truth camera poses.

4. Experiments

We report evaluation results for quality of novel view synthesis and cross-dataset generalization, as well as pose estimation on several datasets.

4.1. Experimental Settings

Dataset. We train and evaluate our method on the RealEstate10K (RE10K) [53] which contains large-scale real estate videos from YouTube, and ACID [28] dataset

which features nature scenes captured by aerial drones. Camera poses for both datasets are obtained via SfM, and we follow the official train-test split used in prior works [4, 8, 49]. Following [49], we evaluate our method under varying camera overlaps, categorizing input pairs based on image overlap ratios: small (0.05%–0.3%), medium (0.3%–0.55%), and large (0.55%–0.8%), determined using a pretrained dense image matching method [12]. To study the impact of training data size, we incorporate the DL3DV dataset [27], an outdoor dataset with 10K videos and diverse camera motions beyond RE10K. For cross-dataset generalization, we follow [8, 49] and evaluate on the object-centric DTU dataset [19].

Baselines. For novel view synthesis, we compare to baselines including pose-required methods (pixelSplat [4] and MVsplat [8]), supervised pose-free methods (CoPoNeRF [17], Splatt3R [37], and NoPoSplat [49]) and self-supervised methods (SelfSplat [21] and PF3plat [16]). For pose estimation, we compare to Superpoint [9] + SuperGlue [35], DUST3R [45], MAST3R [23], and splatting-based methods including NoPoSplat, SelfSplat and PF3plat.

Evaluation Protocol. We evaluate novel view synthesis with the standard metrics: pixel-level PSNR, patch-level SSIM [46], and feature-level LPIPS [52]. As in previous works [35, 49], for pose estimation, we report the area under the cumulative pose error curve (AUC) at thresholds of 5° , 10° , 20° , where the pose error is the maximum of the angular errors in rotation and translation.

During evaluation for NVS, the target images are typically rendered using ground-truth poses [4, 8, 17, 37]. A different strategy is to render novel views using the estimated target poses, as in PF3plat [16] and SelfSplat [21]. Alternatively, NoPoSplat [49] employs an evaluation-time pose alignment (EPA) strategy, which optimizes the target pose during evaluation while keeping the reconstructed Gaussians frozen, such that the rendered image closely matches the ground truth. Unless otherwise specified, we use estimated poses for a comprehensive evaluation and also report EPA results for a fair comparison with NoPoSplat. EPA decouples rendering quality from pose estimation, allowing direct assessment of Gaussian reconstruction. In contrast, rendering with estimated poses jointly evaluates both reconstruction quality and the alignment between the estimated poses and the learned Gaussians.

4.2. Implementation Details.

Our method is implemented in PyTorch, using a CUDA-based 3DGS renderer with gradient support for camera poses. All models are trained on a single A100 GPU. The encoder follows a ViT-Large architecture with a patch size of 16, and the decoder is ViT-Base. The encoder, decoder, and Gaussian center head are initialized with pretrained MAST3R [23] weights. The pose head is initialized

| Method | Small | | | Medium | | | Large | | | Average | | | Time (s) |
|----------------------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|----------|
| | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | |
| <i>Pose-Required</i> | | | | | | | | | | | | | |
| pixelSplat | 20.277 | 0.719 | 0.265 | 23.726 | 0.811 | 0.180 | 27.152 | 0.880 | 0.121 | 23.859 | 0.808 | 0.184 | 0.152 |
| MVSplat | 20.371 | 0.725 | 0.250 | 23.808 | 0.814 | 0.172 | 27.466 | 0.885 | 0.115 | 24.012 | 0.812 | 0.175 | 0.059 |
| <i>Supervised Pose-Free</i> | | | | | | | | | | | | | |
| CoPoNeRF | 17.393 | 0.585 | 0.462 | 18.813 | 0.616 | 0.392 | 20.464 | 0.652 | 0.318 | 18.938 | 0.619 | 0.388 | - |
| Splatt3R | 17.789 | 0.582 | 0.375 | 18.828 | 0.607 | 0.330 | 19.243 | 0.593 | 0.317 | 18.688 | 0.337 | 0.596 | 0.042 |
| NoPoSplat* | 22.514 | 0.784 | 0.210 | 24.899 | 0.839 | 0.160 | 27.411 | 0.883 | 0.119 | 25.033 | 0.838 | 0.160 | 0.042 |
| <i>Self-Supervised Pose-Free</i> | | | | | | | | | | | | | |
| SelfSplat | 14.828 | 0.543 | 0.469 | 18.857 | 0.679 | 0.328 | 23.338 | 0.798 | 0.208 | 19.152 | 0.680 | 0.328 | 0.101 |
| PF3plat | 18.358 | 0.668 | 0.298 | 20.953 | 0.741 | 0.231 | 23.491 | 0.795 | 0.179 | 21.042 | 0.739 | 0.233 | 1.171 |
| SPFSplat | <u>22.897</u> | <u>0.792</u> | <u>0.201</u> | <u>25.334</u> | <u>0.847</u> | <u>0.153</u> | <u>27.947</u> | <u>0.894</u> | 0.110 | <u>25.484</u> | <u>0.847</u> | <u>0.153</u> | 0.044 |
| SPFSplat* | 23.178 | 0.796 | 0.200 | 25.695 | 0.853 | 0.151 | 28.377 | 0.899 | <u>0.111</u> | 25.845 | 0.852 | 0.152 | 0.044 |

Table 1. Performance comparison of novel view synthesis on the RE10K dataset [53]. The reported runtime reflects only the time required to reconstruct 3D Gaussians from two input images. Our method achieves computational efficiency comparable to NoPoSplat while significantly outperforming previous state-of-the-art pose-required and pose-free methods across all overlap settings, especially in low-overlap scenes. The **best** and second-best results are highlighted. * indicates the use of evaluation-time pose alignment (EPA) strategy.

| Method | Small | | | Medium | | | Large | | | Average | | |
|----------------------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|
| | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow |
| <i>Pose-Required</i> | | | | | | | | | | | | |
| pixelSplat | 22.088 | 0.655 | 0.284 | 25.525 | <u>0.777</u> | 0.197 | 28.527 | 0.854 | 0.139 | 25.889 | 0.780 | 0.194 |
| MVSplat | 21.412 | 0.640 | 0.290 | 25.150 | 0.772 | 0.198 | 28.457 | 0.854 | 0.137 | 25.561 | 0.775 | 0.195 |
| <i>Supervised Pose-Free</i> | | | | | | | | | | | | |
| CoPoNeRF | 18.651 | 0.551 | 0.485 | 20.654 | 0.595 | 0.418 | 22.654 | 0.652 | 0.343 | 20.950 | 0.606 | 0.406 |
| Splatt3R | 17.419 | 0.501 | 0.434 | 18.257 | 0.514 | 0.405 | 18.134 | 0.508 | 0.395 | 18.060 | 0.510 | 0.407 |
| NoPoSplat* | <u>23.087</u> | <u>0.685</u> | <u>0.258</u> | <u>25.624</u> | <u>0.777</u> | 0.193 | 28.043 | 0.841 | 0.144 | 25.961 | <u>0.781</u> | 0.189 |
| <i>Self-Supervised Pose-Free</i> | | | | | | | | | | | | |
| SelfSplat | 18.301 | 0.568 | 0.408 | 21.375 | 0.676 | 0.314 | 25.219 | 0.792 | 0.214 | 22.089 | 0.694 | 0.298 |
| PF3plat | 18.112 | 0.537 | 0.376 | 20.732 | 0.615 | 0.307 | 23.607 | 0.710 | 0.228 | 21.206 | 0.632 | 0.293 |
| SPFSplat | 22.667 | 0.665 | 0.262 | 25.620 | 0.773 | <u>0.192</u> | <u>28.607</u> | <u>0.856</u> | <u>0.136</u> | <u>26.070</u> | <u>0.781</u> | <u>0.186</u> |
| SPFSplat* | 23.676 | 0.708 | 0.243 | 26.351 | 0.801 | 0.182 | 29.170 | 0.870 | 0.131 | 26.796 | 0.807 | 0.176 |

Table 2. Performance comparison of novel view synthesis on the ACID dataset [28]. The **best** and second best results are highlighted.

to approximate the identity rotation matrix for stable convergence. All remaining layers are randomly initialized. The loss weights for LPIPS and reprojection loss are set to 0.05 and 0.001, respectively. All experiments are conducted at 256×256 resolution.

4.3. Results

Novel View Synthesis. We present quantitative results in Tab. 1 and Tab. 2. Our model outperforms all SOTA methods, including pose-required and supervised pose-free approaches. Notably, it achieves superior results even in cases of small input image overlap and extreme viewpoint changes, as illustrated in Fig. 3, despite the fact that no ground-truth poses were used during training. Furthermore, even without evaluation-time pose alignment (EPA), our model still surpasses NoPoSplat, which indicates that our estimated poses are well aligned with the Gaussians. We report a reconstruction time of 0.044 seconds for 3D Gaussians from two 256×256 input images, making it approximately $3.5 \times$ and $27 \times$ faster than pixelSplat and PF3plat, respectively, on the same A6000 GPU. SPFSplat achieves such efficiency through a shared backbone for both pose and

Gaussian prediction, with a lightweight MLP-based pose head. In contrast, PF3plat employs separate modules and relies on computationally expensive local feature matching for pose estimation.

Relative Pose Estimation. We evaluate pose estimation between two input images on RE10K and ACID, as shown in Tab. 3. The evaluation details of baselines are provided in the appendix. Our method supports two pose estimation strategies: direct regression via the pose head, and estimation via PnP [15] with RANSAC [13], using the predicted 3D Gaussian centers. Both yield similarly strong results, indicating accurate alignment between the estimated poses and the reconstructed 3D points. Notably, despite being trained without geometry priors, our method significantly outperforms recent approaches, including MAST3R, from which our model is initialized, demonstrating that our framework effectively optimizes both camera poses and 3D structure using only image-level supervision.

Cross-Dataset Generalization. To assess zero-shot generalization, we train exclusively on RE10K (indoor scenes) and evaluate on ACID (outdoor scenes) and DTU (object-centric scenes). The results in Tab. 4 and Fig. 4 demonstrate

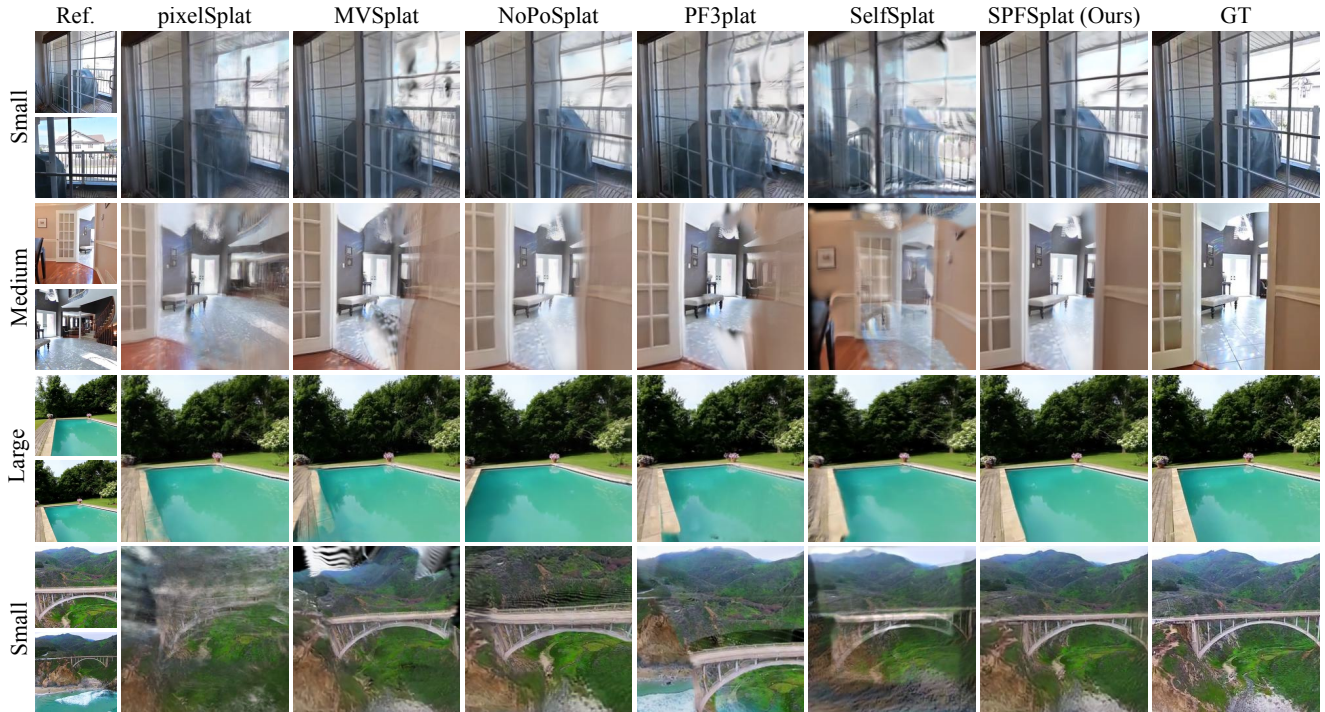


Figure 3. Qualitative comparison on RE10K (top three rows) and ACID (bottom row). Compared to baselines, our method 1) reduces misaligned blending artifacts and ghosting effects, 2) better handles extreme viewpoint changes and texture-less areas (e.g. window), and 3) preserves overall scene geometry (e.g. bridge) and finer details (e.g. swimming pool).



(a) Cross-Dataset Generalization: RE10K \rightarrow ACID

(b) Cross-Dataset Generalization: RE10K \rightarrow DTU

Figure 4. Cross-dataset generalization. Some failure regions are highlighted by red rectangles for visual reference.

| Method | RE10K | | | ACID | | |
|-----------------------|---------------|----------------|----------------|---------------|----------------|----------------|
| | 5° \uparrow | 10° \uparrow | 20° \uparrow | 5° \uparrow | 10° \uparrow | 20° \uparrow |
| SP + SG | 0.234 | 0.406 | 0.569 | 0.228 | 0.363 | 0.500 |
| DUST3R | 0.336 | 0.541 | 0.702 | 0.118 | 0.279 | 0.470 |
| MASt3R | 0.281 | 0.494 | 0.671 | 0.138 | 0.312 | 0.507 |
| NoPoSplat | 0.572 | 0.728 | <u>0.833</u> | 0.335 | 0.497 | 0.645 |
| SelfSplat | 0.207 | 0.392 | 0.576 | 0.205 | 0.363 | 0.531 |
| PF3plat | 0.187 | 0.398 | 0.613 | 0.060 | 0.165 | 0.340 |
| SPFSplat (PnP) | <u>0.613</u> | <u>0.754</u> | 0.845 | <u>0.355</u> | <u>0.516</u> | <u>0.658</u> |
| SPFSplat | 0.617 | 0.755 | 0.845 | 0.364 | 0.520 | 0.662 |

Table 3. Pose estimation performance in AUC with various thresholds on RE10K and ACID datasets. To assess generalizability, we evaluate on ACID using the models trained only on RE10K for all splat-based methods. Our method achieves the best results in both in-domain and out-of-domain settings.

that our approach outperforms all SOTA methods. With no ground-truth poses used during training, our model learns

| Method | ACID | | | DTU | | |
|------------------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|
| | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow |
| pixelSplat | 25.477 | 0.770 | 0.207 | 15.067 | 0.539 | 0.341 |
| MVsplat | 25.525 | 0.773 | 0.199 | 14.542 | 0.537 | 0.324 |
| NoPoSplat* | 25.765 | 0.776 | 0.199 | 17.899 | <u>0.629</u> | 0.279 |
| SelfSplat | 22.204 | 0.686 | 0.316 | 13.249 | 0.434 | 0.441 |
| PF3plat | 20.726 | 0.610 | 0.308 | 12.972 | 0.407 | 0.464 |
| SPFSplat | <u>25.965</u> | <u>0.781</u> | <u>0.190</u> | 16.550 | 0.579 | <u>0.270</u> |
| SPFSplat* | 26.697 | 0.806 | 0.181 | 18.297 | 0.660 | 0.255 |

Table 4. Cross-dataset generalization. All methods are trained on RE10K and evaluated in a zero-shot setting on ACID and DTU. Our method demonstrates superior generalization compared to SOTA approaches, even outperforming NoPoSplat’s ACID-trained model (PSNR: 25.961) as reported in Tab. 2.

to align the Gaussians with the predicted poses, enabling strong generalization to out-of-distribution scenes.

Geometry Reconstruction. As shown in Fig. 5, our SPFS-

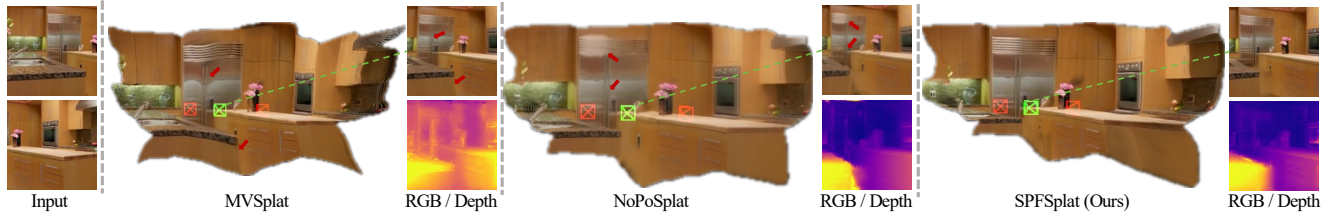


Figure 5. Comparison of 3D Gaussians and rendered results. Input and target camera poses are shown in red and green, respectively. Rendered images and depth maps are displayed on the right. Our method produces higher-quality 3D Gaussians and achieves superior rendering compared to baseline methods. Some regions with distorted or incorrect geometry are highlighted with red arrows.

plat generates higher-quality 3D Gaussian primitives than baseline methods, despite being trained without ground-truth poses. The reconstructed structures are sharper and more accurate, indicating better Gaussian alignment across views. This improvement mainly stems from jointly optimizing Gaussians and camera poses, which encourages a stronger understanding of scene geometry.

4.4. Ablation Analysis

Ablation on Different Components. We conduct an ablation study to assess the contribution of each component in our method, as shown in Tab. 5. As seen from (a) to (b), removing intrinsic embeddings from the backbone slightly reduces performance due to scale ambiguity in both 3D Gaussian learning and pose estimation. However, even without intrinsic embeddings, our method still outperforms NoPoSplat with intrinsic embeddings (PSNR: 25.033). Comparing (a) and (c), removing the reprojection loss while retaining only the image rendering loss on target images significantly degrades both NVS and pose estimation performance, highlighting the importance of geometric constraints between 3D points and camera poses for accurate reconstruction.

| Method | NVS* | | | Pose | | |
|----------------------|-----------------|-----------------|--------------------|-----------------------|------------------------|------------------------|
| | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | 5 $^\circ$ \uparrow | 10 $^\circ$ \uparrow | 20 $^\circ$ \uparrow |
| (a) SPFSplat (Ours) | 25.845 | 0.852 | 0.152 | 0.617 | 0.755 | 0.845 |
| (b) w/o intrin. emb. | 25.519 | 0.844 | 0.156 | 0.562 | 0.717 | 0.823 |
| (c) w/o reproj. loss | 21.914 | 0.742 | 0.251 | 0.028 | 0.102 | 0.263 |
| (d) w/ gt pose loss | 25.910 | 0.860 | 0.150 | 0.691 | 0.810 | 0.885 |

Table 5. Component ablations on RE10K. * indicates the use of EPA strategy. See the appendix for NVS results without EPA.

Ablation on Ground-truth Poses. To evaluate our method’s ability to reconstruct geometry without pose priors, we introduce a pose loss during training that minimizes the difference between predicted and ground-truth poses (details provided in the appendix). This supervision is used only during training, keeping the method pose-free at inference. As shown in Tab. 5 (a) to (d), pose supervision improves pose accuracy but has only a marginal effect on NVS performance, highlighting our model’s strong capacity to reconstruct geometry without explicit pose supervision. It also suggests that NVS quality depends on factors beyond pose accuracy: challenges such as occlusion, texture-less

regions, and extreme viewpoint changes may require generative abilities or explicit 3D supervision.

Scale to More Training Data. Since our approach does not require ground-truth poses for training, it can scale efficiently to larger training datasets with minimal additional cost. To assess the impact of training data size, we train on a combination of RE10K and DL3DV. As shown in Tab. 6, increasing the amount of training data improves performance on both RE10K and ACID. This improvement is likely due to the increased diversity of camera motions in DL3DV, which enhances the model’s ability to generalize across different viewing conditions.

| Training data | RE10K | | | ACID | | |
|---------------|-----------------------|------------------------|------------------------|-----------------------|------------------------|------------------------|
| | 5 $^\circ$ \uparrow | 10 $^\circ$ \uparrow | 20 $^\circ$ \uparrow | 5 $^\circ$ \uparrow | 10 $^\circ$ \uparrow | 20 $^\circ$ \uparrow |
| Re10K | 0.617 | 0.755 | 0.845 | 0.364 | 0.520 | 0.662 |
| Re10K + DL3DV | 0.635 | 0.768 | 0.852 | 0.395 | 0.544 | 0.680 |

Table 6. Ablation on training data size.

Extension to Multiple Views. Our method extends naturally to multiple input views. As shown in Table 7, quantitative results demonstrate that NVS performance improves consistently with an increasing number of context views.

| Num of Views | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow |
|--------------|-----------------|-----------------|--------------------|
| 2 views | 25.403 | 0.845 | 0.154 |
| 3 views | 26.724 | 0.871 | 0.128 |
| 5 views | 26.891 | 0.875 | 0.122 |
| 10 views | 27.159 | 0.880 | 0.115 |

Table 7. Novel view synthesis with varying input view numbers.

5. Conclusion

This paper introduces SPFSplat, an efficient self-supervised pose-free framework designed for sparse-view 3D reconstruction. It employs a shared backbone to simultaneously predict 3D Gaussian representations and camera poses in a canonical space given unposed input views. A reprojection loss is also incorporated with the conventional rendering loss to enhance geometric alignment. Experimental evaluations highlight the superior performance of SPFSplat in novel view synthesis, relative pose estimation, and zero-shot generalization. Notably, the independence from ground-truth pose annotations underscores its potential for scalable training on large-scale real-world data.

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. 3
- [2] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5044–5053, 2023. 4
- [3] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. In *European Conference on Computer Vision*, pages 421–440. Springer, 2024. 5
- [4] 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. 1, 2, 3, 4, 5, 12
- [5] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14124–14133, 2021. 1, 2, 3
- [6] Shuai Chen, Tommaso Cavallari, Victor Adrian Prisacariu, and Eric Brachmann. Map-relative pose regression for visual re-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20665–20674, 2024. 4
- [7] Yu Chen and Gim Hee Lee. Dbarf: Deep bundle-adjusting generalizable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24–34, 2023. 1, 3
- [8] 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. In *European Conference on Computer Vision*, pages 370–386. Springer, 2024. 1, 2, 3, 4, 5, 12
- [9] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabbinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018. 3, 5
- [10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 4
- [11] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 3
- [12] Johan Edstedt, Qiyu Sun, Georg Bökman, Márten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19790–19800, 2024. 5
- [13] MA FISCHLER AND. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 6, 12
- [14] 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*, pages 20796–20805, 2024. 3
- [15] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 3, 6, 12
- [16] Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jisang Han, Jiaolong Yang, Chong Luo, and Seungryong Kim. Pf3plat: Pose-free feed-forward 3d gaussian splatting. *arXiv preprint arXiv:2410.22128*, 2024. 2, 3, 4, 5
- [17] Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jiaolong Yang, Seungryong Kim, and Chong Luo. Unifying correspondence pose and nerf for generalized pose-free novel view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20196–20206, 2024. 1, 3, 5
- [18] Ranran Huang, Jiancheng Cai, Chao Li, Zhuoyuan Wu, Xinmin Liu, and Zhenhua Chai. Drkf: Distilled rotated kernel fusion for efficient rotation invariant descriptors in local feature matching. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1885–1892. IEEE, 2023. 3
- [19] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 5
- [20] Hanwen Jiang, Zhenyu Jiang, Yue Zhao, and Qixing Huang. LEAP: Liberate sparse-view 3d modeling from camera poses. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 3
- [21] Gyeongjin Kang, Jisang Yoo, Jihyeon Park, Seungtae Nam, Hyeonsoo Im, Sangheon Shin, Sangpil Kim, and Eunbyung Park. Selfsplat: Pose-free and 3d prior-free generalizable 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22012–22022, 2025. 2, 3, 4, 5
- [22] 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, 2
- [23] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Confer-*

- ence on Computer Vision, pages 71–91. Springer, 2024. 2, 3, 4, 5
- [24] Amy Lin, Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose++: Recovering 6d poses from sparse-view observations. In *2024 International Conference on 3D Vision (3DV)*, pages 106–115. IEEE, 2024. 3
- [25] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5741–5751, 2021. 1, 3
- [26] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17627–17638, 2023. 3
- [27] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. D13dv-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. 5
- [28] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14458–14467, 2021. 5, 6
- [29] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2
- [30] Bharath Raj Nagoor Kani, Hsin-Ying Lee, Sergey Tulyakov, and Shubham Tulsiani. Upfusion: Novel view diffusion from unposed sparse view observations. In *European Conference on Computer Vision (ECCV)*, 2024. 1, 3
- [31] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 4
- [32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 3
- [33] Mehdi SM Sajjadi, Henning Meyer, Etienne Pot, Urs Bergmann, Klaus Greff, Noha Radwan, Suhani Vora, Mario Lučić, Daniel Duckworth, Alexey Dosovitskiy, et al. Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6229–6238, 2022. 1, 3
- [34] Seyed Sadegh Mohseni Salehi, Shadab Khan, Deniz Erdogmus, and Ali Gholipour. Real-time deep pose estimation with geodesic loss for image-to-template rigid registration. *IEEE transactions on medical imaging*, 38(2):470–481, 2018. 12
- [35] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 3, 5
- [36] 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, 5
- [37] Brandon Smart, Chuanxia Zheng, Iro Laina, and Victor Adrian Prisacariu. Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. *arXiv preprint arXiv:2408.13912*, 2024. 1, 2, 3, 4, 5
- [38] Cameron Smith, Yilun Du, Ayush Tewari, and Vincent Sitzmann. Flowcam: Training generalizable 3d radiance fields without camera poses via pixel-aligned scene flow. *Advances in Neural Information Processing Systems*, 36:1476–1488, 2023. 3
- [39] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 3
- [40] 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, 2024. 1, 2, 3
- [41] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4190–4200, 2023. 3
- [42] Jianyuan Wang, Christian Ruppert, and David Novotny. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9773–9783, 2023. 3
- [43] Jianyuan Wang, Nikita Karaev, Christian Ruppert, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21686–21697, 2024. 3
- [44] Peng Wang, Hao Tan, Sai Bi, Yinghao Xu, Fujun Luan, Kalyan Sunkavalli, Wenping Wang, Zexiang Xu, and Kai Zhang. PF-LRM: Pose-free large reconstruction model for joint pose and shape prediction. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 3
- [45] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 2, 3, 4, 5
- [46] 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. 5
- [47] Haofei Xu, Anpei Chen, Yuedong Chen, Christos Sakaridis, Yulun Zhang, Marc Pollefeys, Andreas Geiger, and Fisher

- Yu. Murf: multi-baseline radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20041–20050, 2024. [1](#), [2](#), [3](#)
- [48] 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. In *European Conference on Computer Vision*, pages 1–20. Springer, 2024. [1](#), [2](#), [3](#), [4](#)
- [49] Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. In *The Thirteenth International Conference on Learning Representations*, 2025. [1](#), [2](#), [3](#), [4](#), [5](#), [12](#)
- [50] Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *European Conference on Computer Vision*, pages 592–611. Springer, 2022. [3](#)
- [51] 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, 2024. [1](#), [2](#), [3](#), [4](#)
- [52] 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. [4](#), [5](#)
- [53] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018. [5](#), [6](#)
- [54] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5745–5753, 2019. [4](#)