Splat-SAP: Feed-Forward Gaussian Splatting for Human-Centered Scene with Scale-Aware Point Map Reconstruction

Boyao Zhou[2,1], Shunyuan Zheng[2], Zhanfeng Liao[1], Zihan Ma[1]*, Hanzhang Tu[1], Boning Liu[1], Yebin Liu[1]†

[1]Department of Automation, Tsinghua University
[2]Ant Group
zhouboyao.zby@antgroup.com, liuyebin@mail.tsinghua.edu.cn

Abstract

We present Splat-SAP, a feed-forward approach to render novel views of human-centered scenes from binocular cameras with large sparsity. Gaussian Splatting has shown its promising potential in rendering tasks, but it typically necessitates per-scene optimization with dense input views. Although some recent approaches achieve feed-forward Gaussian Splatting rendering through geometry priors obtained by multi-view stereo, such approaches still require largely overlapped input views to establish the geometry prior. To bridge this gap, we leverage pixel-wise point map reconstruction to represent geometry which is robust to large sparsity for its independent view modeling. In general, we propose a two-stage learning strategy. In stage 1, we transform the point map into real space via an iterative affinity learning process, which facilitates camera control in the following. In stage 2, we project point maps of two input views onto the target view plane and refine such geometry via stereo matching. Furthermore, we anchor Gaussian primitives on this refined plane in order to render high-quality images. As a metric representation, the scale-aware point map in stage 1 is trained in a self-supervised manner without 3D supervision and stage 2 is supervised with photo-metric loss. We collect multi-view human-centered data and demonstrate that our method improves both the stability of point map reconstruction and the visual quality of free-viewpoint rendering.

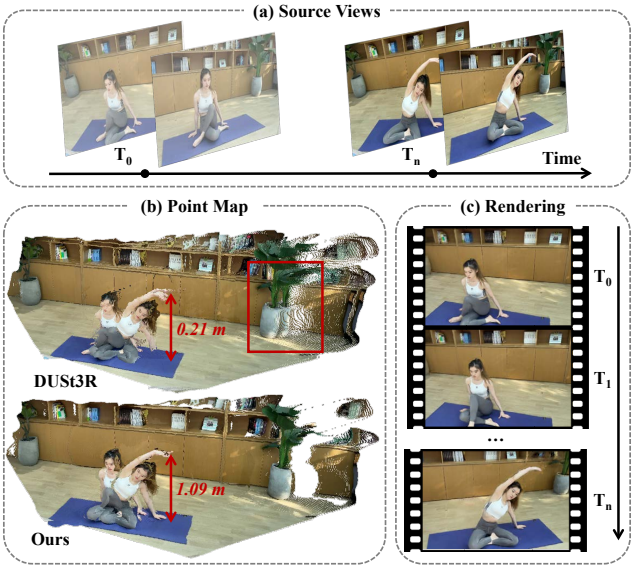


Figure 1: Human-centered scene reconstruction and free-view video synthesis. (a) Source view inputs, (b) our metric scale point map reconstruction, and (c) free-view rendering.

Introduction

Feed-forward free-viewpoint video synthesis is a crucial task, especially in the setting of sparse views, which could serve many downstream applications such as telecommunications, stage/sports broadcasts, and so on. Existing pipelines are typically based on differentiable rendering (Mildenhall et al. 2020; Xu et al. 2022; Yu et al. 2021a; Chen et al. 2021; Lin et al. 2022; Wu et al. 2024; Sun et al. 2024) with the development of neural network. Particularly, Gaussian Splatting (Kerbl et al. 2023) shows its advancement for the high efficiency of rendering and the capable mechanism of back-propagation, but relies on minute-level optimization for each scene and very dense input views.

Recently, Gaussian related methods (Chen et al. 2024; Liu et al. 2024; Charatan et al. 2024) achieve instant inference in a feed-forward manner, avoiding per-scene optimization, for real-time applications such as telecommunication systems (Tu et al. 2024) and human-scene synthesis (Zheng et al. 2024; Zhou et al. 2025). However, a common strategy is to estimate Gaussian primitive maps defined on the source views, leveraging geometry proxies of multi-view stereo (MVS) (Chen et al. 2024; Liu et al. 2024) or binocular stereo-matching (Zheng et al. 2024; Zhou et al. 2025). Such methods require a large overlap of paired images, which increases the redundancy of Gaussian primitives in this overlap area. Otherwise, they could not provide reasonable geometry prior when input cameras are with large sparsity.

More recently, DUSt3R (Wang et al. 2024; Leroy, Cabon, and Revaud 2024) proposes a novel geometry representation as point maps of input views, which assigns each pixel of input images to a free 3D point. Unlike traditional MVS

*Work done during an internship at Tsinghua University.
†Corresponding author.

methods (Yao et al. 2018; Yang et al. 2020), DUSt3R gets rid of stereo constraints and achieves pixel-aligned point maps of binocular inputs from very sparse views, by training on immense 3D geometry data. To alleviate the impact of the high degree of freedom, DUSt3R and its follow-ups (Leroy, Cabon, and Revaud 2024; Wang et al. 2025b) normalize the scale of the reconstructed point map with an average point distance of each scene, which typically causes a dramatic instability of reconstruction in consecutive frames, see Fig. 1(b). Some recent methods (Smart et al. 2024; Ye et al. 2025) leverage point map representation for static scene rendering in canonical space. However, human movement in the scene would cause a relative depth difference in canonical space and lead to large jitters for free-view video rendering, due to the lack of stereo constraint. In addition, training a foundation model for scale-aware geometry typically requires immense 3D data, but it is always tedious and cumbersome to acquire 3D geometry data. Therefore, the key point is to obtain scale-aware geometry in a self-supervised manner and to ease the burden of 3D data acquisition.

In this paper, we propose Splat-SAP to achieve human-centered scene reconstruction in real metric space and feed-forward rendering of novel views via Gaussian plane, when inputting a pair of images and camera calibration. Unlike (Wang et al. 2024; Leroy, Cabon, and Revaud 2024) representing a scale-invariant point map in canonical space, we inject camera intrinsic embedding (Ye et al. 2025) and global image feature (Wang et al. 2025a) into a network as input to learn a **scaling** factor to transform the estimated point map from canonical space to real space. Since the point representation in the original design of DUSt3R (Wang et al. 2024; Leroy, Cabon, and Revaud 2024) lacks stereo constraints between two source views, there always exists misalignment between the two point maps. Thus, we compute the cost between 2 source views to do an iterative ***coarse registration*** of 2 reconstructed point maps, by projecting the feature from one view to another with calibrated camera pose. This registration is in the format of a translation map, denoting pixel-wise shift. Our scaling factor by intrinsic embedding and translation learning by extrinsic projection compose exactly an **affine transformation** of point map from canonical space to real space.

In terms of rendering, we anchor Gaussian primitives directly on the target view as a Gaussian plane, so that reducing the redundancy of using directly two point maps of source views as Gaussian positions (Zheng et al. 2024; Chen et al. 2024; Smart et al. 2024; Ye et al. 2025). Depth of the Gaussian plane is initialized by projecting two point maps via α-blending (Kerbl et al. 2023), which largely eases the burden of accurate depth estimation. Further, we do a ***fine registration*** with strict stereo constraint (Yao et al. 2018; Lin et al. 2022; Liu et al. 2024), which relies on a 3D cost volume by sampling several depth candidates along each pixel ray near the initialized depth. With such 3D cost representation, we can aggregate more 3D information to overcome the unobservation issue due to the large sparsity, and to estimate accurate depth. The color of the Gaussian plane can be initialized by warping source view pixels directly via the estimated depth. Additionally, we incorporate both fine 2D and dense 3D features to estimate Gaussian primitives and refine Gaussian color for high-quality rendering.

More importantly, our pipeline can be trained without geometry supervision, different from (Wang et al. 2024; Leroy, Cabon, and Revaud 2024; Wang et al. 2025b). To this end, we collect large-scale multi-view data of over 10,000 frames of motion sequences of human-centered scenes to train our model. We validate the effectiveness of our method on diverse camera settings, *e.g.* industry camera, mobile phone, and GoPro, for both reconstruction and rendering tasks. In summary, we claim three following contributions:

- We introduce a feed-forward pipeline to reconstruct scale-aware point maps and to render free-view video of human-centered scenes, where the point maps are trained in a self-supervised manner without any 3D supervision.
- We propose a 2D coarse to 3D fine registration strategy to estimate scale-aware point maps with a learnable affinity.
- We design a Gaussian plane, leveraging scale-aware point maps and incorporating both 2D and 3D features, to guarantee the efficiency and completeness of rendering.

Related Work

Novel View Synthesis

Neural Radiance Fields (Mildenhall et al. 2020; Barron et al. 2021) achieve photo-realistic rendering quality by applying volume rendering which aggregates the sampled neural features along the ray. Recently, 3D Gaussian Splatting (Kerbl et al. 2023) has made significant advances in neural rendering for its real-time rendering efficiency. This outstanding technique models static scenes by optimizing a set of Gaussian primitives (Kerbl et al. 2023; Lu et al. 2024; Yu et al. 2024), including properties of position, scaling, rotation, and opacity. Some methods progress further to model dynamic scenes with time-varying Gaussian primitives (Li et al. 2024; Yang et al. 2024; Yan et al. 2024; Sun et al. 2025), with 4D neural representation (Wu et al. 2024; Xu et al. 2024a,b), or with on-the-fly streamable training (Luiten et al. 2024; Sun et al. 2024; Girish et al. 2025; Gao et al. 2025). Although immensely accelerated, the inevitable per-scene optimization still requires minutes to accomplish.

To eliminate the long-time optimization process, generalizable neural rendering methods (Wang et al. 2021b; Yu et al. 2021a; Charatan et al. 2024; Zheng et al. 2024) have been developed for feed-forward novel view rendering. Typically, these generalizable paradigms resort to leveraging the learned 2D/3D priors from extensive data (Zhou et al. 2018; Liu et al. 2021; Yu et al. 2021b) to ease the long-term optimization. In this practice, ENeRF (Lin et al. 2022) integrates cost volume to provide a coarse depth initialization and thus reduce the sampling points, leading to an efficient framework. With respect to Gaussian-based methods (Xu et al. 2025; Charatan et al. 2024; Zheng et al. 2024; Liu et al. 2024), the solution is parallel to that in generalizable NeRF, *e.g.* by using epipolar stereo and cost volume. However, some methods (Zheng et al. 2024) necessitate ground truth depth for training, and others (Charatan et al. 2024; Chen et al. 2024; Zhou et al. 2025) are limited under sparse

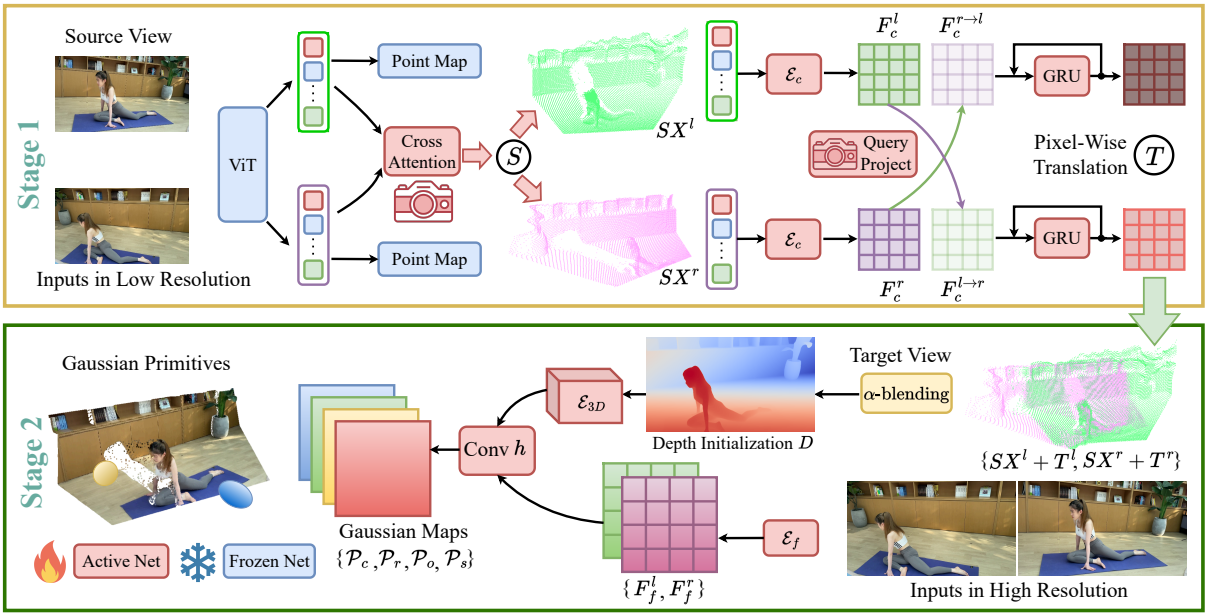


Figure 2: Overview of Splat-SAP. Our method consists of two stages. In the first stage, we take two coarse images as input and predict corresponding point maps, along with an affine transform. In the second stage, our refinement module takes transformed points and fine-resolution images as input, and predicts Gaussian plane of target view for high-quality rendering.

input views, due to the difficulty of establishing correlation with small overlap of input views.

3D Reconstruction

Multi-view stereo is a traditional 3D reconstruction technique, which can be categorized according to the output modality, including point cloud (Lhuillier and Quan 2005; Furukawa and Ponce 2009), volumetric representation (Seitz and Dyer 1999; Kutulakos and Seitz 2000), and depth map (Campbell et al. 2008; Schönberger et al. 2016). MVS-Net (Yao et al. 2018) opens up the era of deep learning-based MVS methods. Binocular Stereo (Zabih and Woodfill 1994), as a special kind of MVS, aims to find the maximum correspondence on the horizontal epipolar line. However, such methods struggle with invisible issues and large sparsity of input views. Progressing further, the neural implicit surface methods (Wang et al. 2021a, 2023; Li et al. 2023), as a variant of neural radiance fields, perform accurate 3D reconstruction with only rendering loss, which avoids collecting 3D properties for training. With the prevalence of 3D Gaussian Splatting in NVS, a body of research (Huang et al. 2024; Dai et al. 2024; Guédon and Lepetit 2024; Lyu et al. 2024) attempts to adapt it to multi-view 3D Reconstruction with flattened or surfel shaped Gaussian primitives. These neural rendering based methods typically rely on dense input views to supervise with rendering loss and long-time optimization.

More recently, DUSt3R(Wang et al. 2024) proposes a novel 3D representation, defining point maps on a pair of source views, aligning a pixel to a free 3D point, bypassing the need for camera poses. In such ill-posed conditions, both DUSt3R and its follow-ups (Leroy, Cabon, and Revaud 2024; Wang et al. 2025b) bound point maps in scale-invariant canonical space. NoPoSplat (Ye et al. 2025) and Splat3R (Smart et al. 2024) incorporate such representation into the rendering pipeline for static scenes. Some followers (Lu et al. 2025; Zhang et al. 2025) point out that when handling dynamic scenes, DUSt3R encounters two limitations: (1) the misaligned background points, and (2) incorrect foreground depth estimation, causing some regions placed in the background. They address these issues with a global test time optimization on the whole video, while we commit to probing a feed-forward solution in this paper.

Method

Given a pair of images and camera calibration, our method reconstructs scale-aware point maps with an affinity learning in the first stage. In the second stage, we project such point maps onto the target view and refine this geometry to anchor Gaussian primitives for the rendering task. An overview of our 2D-coarse-to-3D-fine pipeline is shown in Fig. 2.

Scale-Aware Geometry Reconstruction

Point Map. It is introduced by DUSt3R (Wang et al. 2024) as a novel but scale-invariant representation $X \in [0,1]^{W\times H\times 3}$ of 3D scene, which is associated with corresponding image I of resolution $W\times H$. We apply the follow-up, MASt3R (Leroy, Cabon, and Revaud 2024), to predict two pieces of point map from source views, $i \in \{l, r\}$ for left and right view, to represent the scene. As a coarse stage, we take $W = 512$ and $H = 288$. MASt3R encodes the input images into patches of features F^i with ViT and then decodes them into X^i in canonical space. Without any stereo constraint, it is capable of predicting reasonable geometries from two cameras in a large sparsity, but hard to control

the target view with real camera parameters, leading to jitters when inferring consecutive frames. In the following, we transform point maps from canonical space to real space in an absolute scale with an affinity, in the format of scaling $S \in \mathbb{R}^3$ and translation $T \in \mathbb{R}^{W \times H \times 3}$.

Scaling. It is a global factor related to camera intrinsic parameters such as focal f. In addition, the distance d between two cameras provides a cue of measurement in real space. Thus, we embed them with positional encoding (Mildenhall et al. 2020) PE

$$e = \text{PE}(f, d) \tag{1}$$

We further process the encoded features F with self- Att_s and cross-attention Att_c

$$\begin{aligned} \langle \mathbf{Q}, \mathbf{K}, \mathbf{V} \rangle &= \langle F\mathbf{W}^Q, F\mathbf{W}^K, F\mathbf{W}^V \rangle \\ f_s &= \text{Avg}(Att_s(\mathbf{Q}_l, \mathbf{K}_l, \mathbf{V}_l)) \\ f_c &= \text{Avg}(Att_c(\mathbf{Q}_l, \mathbf{K}_r, \mathbf{V}_r)) \end{aligned} \tag{2}$$

where the average operator is used to extract global information. We use an MLP to compute the scaling factor

$$S = MLP(f_s, f_c, e) \tag{3}$$

We note that the degree of freedom of S is 3 to deal with the distortion of original point map reconstruction of MASt3R.

Translation. Although the point maps could be rescaled to real space by multiplying the scaling factor, there could still exist point-wise shifts due to the lack of stereo constraint in MASt3R. Inspired by the view consistency check (Yan et al. 2020) in MVS, we believe that such a shift depends not only on the features in one view but also on the corresponding features in the other view. Thus, we process the aforementioned ViT features with a lightweight convolutional encoder $\mathcal{E}_c$ to yield feature map $F_c^i = \mathcal{E}(F^i)$. Furthermore, we obtain the feature map $F_c^{j \to i}$ in the other view j, $j \in \{l, r\}$ and $i \neq j$, by first projecting the rescaled points SX^i onto the view j and then querying the corresponding features with bilinear sampling

$$F_c^{j \to i} = \text{Query}(F_c^j, \text{Proj}(SX^i, K^j)) \tag{4}$$

where K is the camera parameter. We follow the idea of iterative updating (Teed and Deng 2020; Lipson, Teed, and Deng 2021) to compute point-wise translation with GRU operator (Cho et al. 2014) by considering feature maps from both views and the position of each point

$$T^i = \text{GRU}(F^i, F^{j \to i}, SX^i) \tag{5}$$

We obtain the position of the point set in real space with the learned affine transform

$$X_t^i = SX^i + T^i \tag{6}$$

Rendering via Gaussian Plane

3D Refinement. When splatting the aforementioned point set to the target view, we still observe inevitable jitters and holes due to the lack of 3D stereo constraint, see Fig. 6(a). Given the paired images I_f in fine resolution of $W = 1024$ and $H = 576$, we encode them with convolutional layers $\mathcal{E}_f$ into $F_f^i = \mathcal{E}_f(I_f^i)$. Additionally, we project the transformed point set X_t onto the target view k with α-blending mechanism in Gaussian Splatting to yield the initial depth map $\mathcal{D}^k$. For each pixel (u, v), we sample several position candidates $\{d_1, d_2, ...d_N\}$ near the initial depth value $d = \mathcal{D}(u, v)$ along camera ray. For each candidate $d(u, v, n)$, we warp the feature from the source views to the target view

$$p^k(u, v, n) = \text{Proj}^{-1}(d^k(u, v, n), K^k) \tag{7}$$

$$F_f^{i \to k}(u, v, n) = \text{Query}(F_f^i, \text{Proj}(p^k(u, v, n), K^i)) \tag{8}$$

where the warping process can be efficiently achieved by matrix operation. We further process the aggregation of features with 3D convolutions $\mathcal{E}_{3D}$ into a feature volume

$$\Phi^k = \mathcal{E}_{3D}(F_f^{l \to k}, F_f^{r \to k}) \tag{9}$$

Following ENeRF (Lin et al. 2022), we compute the depth probability distribution w_n along the camera ray by regressing with the feature volume Φ. The final position of Gaussian primitives can be represented with the refined depth $\bar{d} = \Sigma_n w_n d_n$.

Gaussian Plane. Once the position of Gaussian is determined, Gaussian plane $\mathcal{G}$ consists of four attribute maps of color, rotation, scaling and opacity

$$\mathcal{G} = \{\mathcal{P}_c, \mathcal{P}_r, \mathcal{P}_s, \mathcal{P}_o\} \tag{10}$$

Using the warping process in Eq. 8, we obtain color $\{C^{l \to k}, C^{r \to k}\}$ of target view warped from source views. We further query the feature ϕ from feature volume Φ for Gaussian primitives via tri-linear interpolation. We thus learn a weighted color to initialize the Gaussian color

$$w_c^i = MLP_c(f_f^l, f_f^r, \phi) \tag{11}$$

$$C^k = \Sigma_i w_c^i C^{i \to k} \tag{12}$$

We arrange all features into the format of a 2D map and further aggregate them into the feature map $\mathcal{M} = \text{Agg}\{f_f^l, f_f^r, \phi\}$. Following GPS-Gaussian, we yield rotation, scaling, and opacity map via convolutional heads $h_a, a = \{r, s, o\}$, considering the Gaussian position Y

$$\mathcal{P}_a = h_a(\mathcal{M}, Y) \tag{13}$$

In addition, we update the initial color with a learned residual color map

$$\Delta C = h_c(\mathcal{M}, Y, C) \tag{14}$$

$$\mathcal{P}_c = \alpha C + (1 - \alpha)\Delta C \tag{15}$$

Finally, we splat the Gaussian plane $\mathcal{G}$ in a fine resolution of 1024×576 to render an image $\hat{I}$ in a higher resolution of 1280×720.

Training

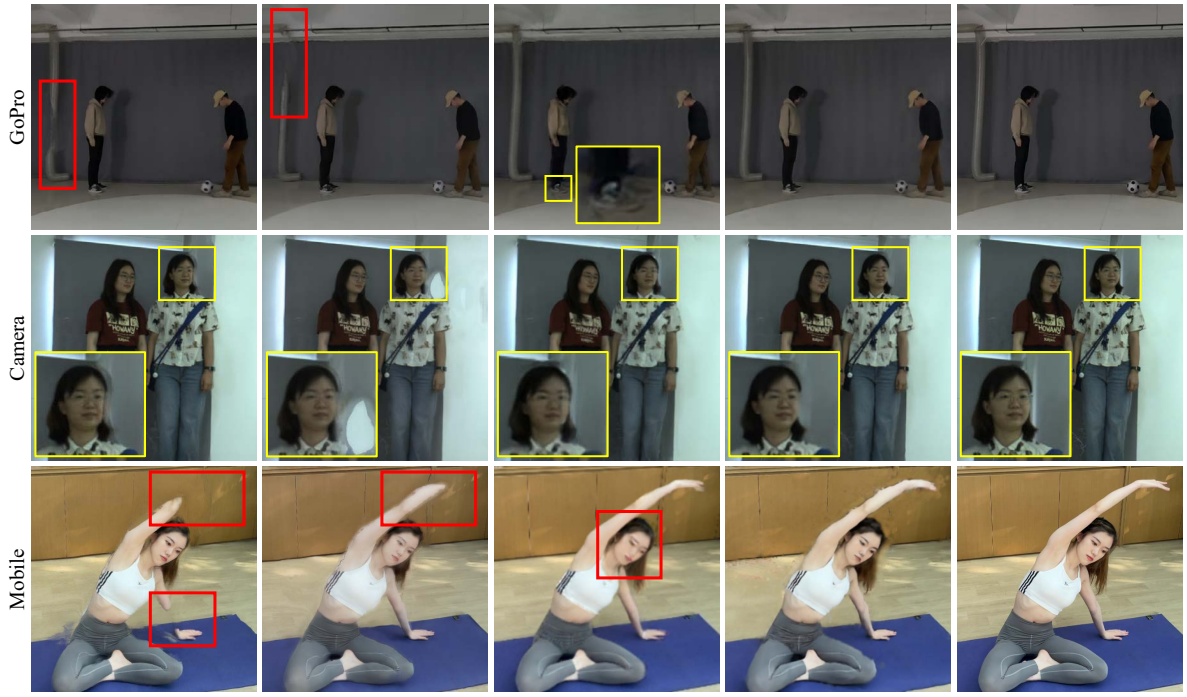
We define the rendering loss as a combination of L1 loss $\mathcal{L}_1$ and SSIM loss (Wang et al. 2004) $\mathcal{L}_{ssim}$

$$\mathcal{L}_{render}(\hat{I}, I^{gt}) = \beta_1 \mathcal{L}_1 + \beta_2 \mathcal{L}_{ssim} \tag{16}$$

where $\hat{I}$ and I^{gt} stand for rendering image and ground truth image. Since the transformed geometry largely impacts the rendering module, we propose a 2-stage training strategy.

Method	Camera			GoPro			Mobile		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NoPoSplat	25.035	0.866	0.173	26.128	0.889	0.121	21.594	0.591	0.272
4D-GS	27.814	0.906	0.150	27.244	0.907	0.205	25.655	0.825	0.284
MVSplat	27.899	0.902	0.148	29.942	0.934	0.157	**26.545**	0.805	0.314
MVSGaussian	29.326	0.957	**0.069**	27.413	0.926	0.151	19.927	0.683	0.272
ENeRF	28.272	0.943	0.084	29.906	0.943	0.108	20.579	0.640	0.302
Ours	**32.220**	**0.957**	0.079	**31.640**	**0.955**	**0.096**	25.721	**0.827**	**0.244**

Table 1: Quantitative comparison of rendering methods on multi-view datasets. Bold highlights the top-performing method, while underline indicates suboptimal performance across various evaluation criteria.



(a) ENeRF (b) MVSGaussian (c) MVSplat (d) Ours (e) Ground Truth

Figure 3: Quantitative comparison of rendering. We show results of (a) ENeRF (Lin et al. 2022), (b) MVSGaussian (Liu et al. 2024), (c) MVSplat (Chen et al. 2024), (d) Ours and (e) Ground Truth for GoPro, Camera and Mobile datasets.

Stage 1. We firstly train the affine transformed point maps with captured multi-view images in a self-supervised manner without any 3D supervision. To this end, we predict Gaussian planes $\mathcal{G}^i = \{\mathcal{P}_p^i, \mathcal{P}_c^i, \mathcal{P}_r^i, \mathcal{P}_s^i, \mathcal{P}_o^i\}, i = \{l, r\}$ on source views. Among them, the position and color plane can be obtained with point map X_t^i and input image I^i. We further use the auxiliary layers $\bar{h}_a$, similar to convolutional operator h_a in Eq. 13, to predict $\mathcal{P}_a^i, a = \{r, s, o\}$. Once the auxiliary Gaussian planes are done, we can render the image on the target view. Inspired by GPS-Gaussian+ (Zhou et al. 2025), we propose a regularization term as Chamfer distance between two 6-dimensional point sets P^l, P^r

$$CD(i \to j) = \frac{1}{|P^i|} \sum_{p_i \in P^i} \min_{p_j \in P^j} \|p_i - p_j\|_2$$
$$\mathcal{L}_{CD} = CD(l \to r) + CD(r \to l) \tag{17}$$

where p_i is pixel-wise point on point map $X_t^i(u, v)$ associated with the corresponding pixel color $I^i(u, v)$. Such a regularization term allows two pieces of point maps to converge to a better geometry.

Therefore, we supervise the affinity learning and auxiliary layers with rendering loss and the regularization term

$$\mathcal{L}_{stage1} = \mathcal{L}_{render} + \gamma \mathcal{L}_{CD} \tag{18}$$

Note that during the training, we freeze the weight of the MASt3R (Leroy, Cabon, and Revaud 2024) network and no longer require the geometry ground truth.

Stage 2. The scale-aware point maps in the previous step allow us to initialize the depth of the target view, which largely improves the stability of the training process in stage 2. Specifically, we train the 3D refinement module and Gaussian planes with photo-metric loss. In practice, we have an

initial color plane $\hat{I}_f = C$ in fine resolution and a splatting image $\hat{I}_h$ in a higher resolution. So the training loss is the combination of two rendering losses

$$\mathcal{L}_{stage2} = \lambda_1 \mathcal{L}_{render}(\hat{I}_f, I_f^{gt}) + \lambda_2 \mathcal{L}_{render}(\hat{I}_h, I_h^{gt}) \quad (19)$$

For both stages 1 and 2, we do not require 3D geometry supervision, which facilitates the training process on real captured 2D images.

Experiment

Settings

Data. We collect multi-view data from 3 types of cameras, including industry camera (THumanMV (Zhou et al. 2025)), mobile phone (4K4D (Xu et al. 2024a) and SelfCap (Xu et al. 2024b)) and our captured GoPro data. For training, we take 15 training sequences of industry camera data, 6 sequences of GoPro data, and around 3000 frames from 4K4D dance sequence and SelfCap yoga sequence. Compared to public datasets, we capture large scenes accommodating sports movement of multi-person with a portable GoPro system. To evaluate our Splat-SAP, we take the sequences of unseen characters or of unseen motions from each dataset. In particular, we train only one model of our affinity module for stage 1, while we train one refinement module per camera type. We pick 6 cameras, facing the characters. The leftmost and the rightmost cameras are fed into the network as source input views. The other 4 views are used as supervision during training and to compute metrics during evaluation.

Metrics. For rendering, the quality of synthesized images is measured with widely used PSNR, SSIM (Wang et al. 2004) and LPIPS (Zhang et al. 2018). We apply Chamfer distance of both directions to evaluate the quality of geometry. We note that the ground truth point set is reconstructed by using Structure-from-Motion (Schonberger and Frahm 2016) with all 6 views under a long-time optimization.

Implementation Details. We employ a two-stage training strategy. We first train the affinity learning module for $100k$ iterations with full training data. For each camera type, we further train the rendering module for $60k$ iterations in stage 2. Our networks can be trained on a single RTX 3090 GPU with 24GB. We set $\alpha = 0.8$ in Eq. 15, $\beta_1 = 0.8$, $\beta_2 = 0.2$ in Eq. 16, $\gamma = 0.5$ in Eq. 18, and $\lambda_1 = 0.5$, $\lambda_2 = 0.5$ in Eq. 19. The input to the first stage is in coarse resolution of $W = 512, H = 288$, while the fine resolution of $W = 1024, H = 576$ for the second stage. We render the high-resolution image of $W = 1280, H = 720$ in the end.

Results

Baselines. We compare Splat-SAP with state-of-the-art methods of feed-forward rendering, including NeRF-based method ENeRF (Lin et al. 2022), as well as Gaussian-based methods MVSplat (Chen et al. 2024), MVSGaussian (Liu et al. 2024) and NoPoSplat (Ye et al. 2025). In addition, we compare with the optimization-based method 4D-GS (Wu et al. 2024), which requires a long time optimization on sequential data. We train ENeRF, MVSplat and MVSGaussian from scratch with the same data setting as our second stage training. We take the pretrained checkpoint of NoPoSplat provided by the original authors and fine-tune it with our training data. We feed the fine-resolution inputs to ENeRF and MVSGaussian, while the coarse ones to MVSplat and NoPoSplat due to the high memory cost.

Figure 4: Qualitative comparison of rendering on a sequence of data. Our method preserves temporal and view consistency against 4D-GS and NoPoSplat.

For geometry, we compare with scale-invariant methods such as DUSt3R (Wang et al. 2024), MASt3R (Leroy, Cabon, and Revaud 2024) and VGGT (Wang et al. 2025a). Furthermore, the scale-aware method Pow3R (Jang et al. 2025) and metric depth estimation method Prompt-DA (Lin et al. 2025) are considered for comparison. Similar to us, Pow3R requires camera calibration as an auxiliary input. Particularly, Prompt-DA feeds images along with corresponding coarse depth maps to the network as inputs.

Rendering Comparisons. We report the quantitative results on datasets of Camera, GoPro and Mobile Phone in Tab. 1. Our method, in general, outperforms others, especially on Camera and GoPro datasets. Since LPIPS (Zhang et al. 2018) is sensitive to higher resolution, our rendering in the resolution of $W = 1280$ is on par with the results of MVSGaussian in the resolution of $W = 1024$ on Camera data. However, MVSGaussian and ENeRF can not handle thin structures and result in some missing parts in Fig. 3(a,b), due to the large sparsity. For mobile data, mobile phones are in the mode of alternate zoom-in and zoom-out, which increases the difficulty for feed-forward methods. Although the setting is tough, our method still renders the fine-grained results with respect to MVSplat in Fig. 3. Due to the lack of geometry regularization term in Eq. 17, two pieces of Gaussians are sometimes misaligned for MVSplat and NoPoSplat in Fig. 3(c) and 4(b) in the case of sparse input views.

NoPoSplat (Ye et al. 2025) also leverages MASt3R as

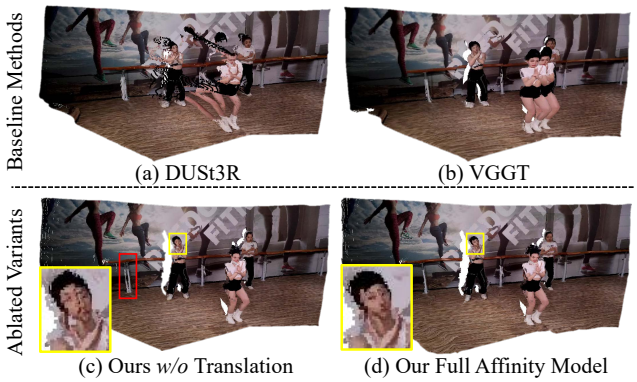


Figure 5: Qualitative comparison of geometry. We show point maps of (a) DUSt3R, (b) VGGT, (c) Ours without pixel-wise translation, and (d) Our full affinity.

a geometry prior, but it gets rid of the traditional stereo constraint, leading to a bad perspective on target view in Fig. 4(b). During the inference, NoPoSplat still requires source and target view camera poses and normalizes them into a relative scale. In Fig. 4(b), such normalization leads to rendering jitters in the case of dynamic differences caused by human movement. 4D-GS (Wu et al. 2024) also neglects geometric constraint and struggles to achieve temporal consistency, see Fig. 4(a), for fast motion under sparse views, even if it optimizes on the sequential data for a long time.

Geometry Comparisons. As scale-invariant methods, the original point maps of DUSt3R, MASt3R and VGGT are defined in canonical space. Therefore, we employ the ground truth scale factor by comparing their bounding box with that of the ground truth to rescale them into real space. However, DUSt3R sometimes immerses into a local minimum, and projects foreground points onto background, see Fig. 5(a). VGGT is not able to handle two-view input with large sparsity, which leads to misalignment of the foreground human from two input views in Fig. 5(b). Further, the scale of the scene can not be perfectly estimated by Pow3R, even using camera calibration, thus causing a large Chamfer distance in Tab. 2. In addition, we feed the rescaled result of MASt3R as coarse depth input to Prompt-DA. But the diffusion-based method increases the uncertainty of prediction, and can not preserve 3D consistency from two input views. Although our method is trained without any geometry loss, we still achieve a superior result in Tab. 2. We notice that the ground truth of geometry is a relatively sparse point cloud when using SfM under 6 input views, thus the Chamfer distance from ground truth to prediction can better reflect geometry quality.

Ablation Study. We first evaluate the effectiveness of our pixel-wise translation in stage 1. Our point maps are rescaled with 3-dimensional scaling factors (Eq. 3) when considering camera intrinsic embedding. The point maps can still not avoid misalignment by only using the scaling operator, see Fig. 5(c). Integrating iterative pixel-wise translation learning, our full affinity module further improves the Chamfer distance in Tab. 2, and corrects misaligned parts in Fig. 5(d).

Figure 6: Qualitative ablation results. Upper row: the rendering comparison between (a) stage 1 and (b) stage 2. Bottom row illustrates the effectiveness of our depth refinement module in stage 2: (c) initial depth map rendered by affine transformed point maps, (d) depth map after refinement, and (e) rendering results.

Method	Pred → GT ↓	GT → Pred ↓
DUSt3R	0.305	0.160
VGGT	0.288	0.129
Pow3R	0.281	0.134
MASt3R	0.212	0.069
Prompt-DA	0.205	0.063
Ours *w/o* Translation	0.191	0.046
Our Full Model	**0.172**	**0.027**

Table 2: Quantitative comparison of geometry.

In addition, we evaluate the effectiveness of our refinement module in stage 2. Alternatively, we can directly synthesize the target view with two Gaussian planes learned by auxiliary layers in stage 1. However, such models typically struggle with holes in the boundary area between foreground and background, see Fig. 6(a). The refinement module can correct artifacts and refine details, see Fig. 6(b, d).

Discussion

Conclusion. We present Splat-SAP, a feed-forward approach for novel view synthesis of human-centered scenes. In particular, we employ a 3D foundation model and utilize iterative affinity learning to reconstruct scale-aware point maps as a coarse geometry. We further leverage geometric constraints to refine the initial geometry, on which we build a Gaussian plane for rendering. The full coarse-to-fine pipeline can be trained with only rendering loss by using multi-view image datasets. Our method achieves superior rendering results in the case of sparse input views.

Limitation. Some floating artifacts occur, as in Fig. 6(e), because MASt3R might predict some floating points on the boundary between foreground and background. Since such regions are observed by only one of two input views, they can not be corrected by our refinement module. We believe that incorporating monocular prior (Xu et al. 2025) would alleviate this problem.

Acknowledgements

This paper is supported by National Key R&D Program of China (2022YFF0902200) and NSFC project (Grants 62125107 and 62301298).

References

Barron, J. T.; Mildenhall, B.; Tancik, M.; Hedman, P.; Martin-Brualla, R.; and Srinivasan, P. P. 2021. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *ICCV*, 5855–5864.

Campbell, N. D.; Vogiatzis, G.; Hernández, C.; and Cipolla, R. 2008. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *ECCV*, 766–779.

Charatan, D.; Li, S. L.; Tagliasacchi, A.; and Sitzmann, V. 2024. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *CVPR*, 19457–19467.

Chen, A.; Xu, Z.; Zhao, F.; Zhang, X.; Xiang, F.; Yu, J.; and Su, H. 2021. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *ICCV*, 14124–14133.

Chen, Y.; Xu, H.; Zheng, C.; Zhuang, B.; Pollefeys, M.; Geiger, A.; Cham, T.-J.; and Cai, J. 2024. MVSplat: Efficient 3D Gaussian Splatting from Sparse Multi-View Images. In *ECCV*, 370–386.

Cho, K.; Van Merriënboer, B.; Bahdanau, D.; and Bengio, Y. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Dai, P.; Xu, J.; Xie, W.; Liu, X.; Wang, H.; and Xu, W. 2024. High-quality surface reconstruction using gaussian surfels. In *SIGGRAPH*, 1–11.

Furukawa, Y.; and Ponce, J. 2009. Accurate, dense, and robust multiview stereopsis. *IEEE TPAMI*, 32(8): 1362–1376.

Gao, Q.; Meng, J.; Wen, C.; Chen, J.; and Zhang, J. 2025. Hicom: Hierarchical coherent motion for dynamic streamable scenes with 3d gaussian splatting. *NeurIPS*, 37: 80609–80633.

Girish, S.; Li, T.; Mazumdar, A.; Shrivastava, A.; De Mello, S.; et al. 2025. QUEEN: QUantized Efficient ENcoding of Dynamic Gaussians for Streaming Free-viewpoint Videos. *NeurIPS*, 37: 43435–43467.

Guédon, A.; and Lepetit, V. 2024. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *CVPR*, 5354–5363.

Huang, B.; Yu, Z.; Chen, A.; Geiger, A.; and Gao, S. 2024. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH*, 1–11.

Jang, W.; Weinzaepfel, P.; Leroy, V.; Agapito, L.; and Revaud, J. 2025. Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. In *CVPR*, 1071–1081.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG*, 42(4): 1–14.

Kutulakos, K. N.; and Seitz, S. M. 2000. A theory of shape by space carving. *IJCV*, 38: 199–218.

Leroy, V.; Cabon, Y.; and Revaud, J. 2024. Grounding image matching in 3d with mast3r. In *ECCV*, 71–91.

Lhuillier, M.; and Quan, L. 2005. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE TPAMI*, 27(3): 418–433.

Li, Z.; Chen, Z.; Li, Z.; and Xu, Y. 2024. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *CVPR*, 8508–8520.

Li, Z.; Müller, T.; Evans, A.; Taylor, R. H.; Unberath, M.; Liu, M.-Y.; and Lin, C.-H. 2023. Neuralangelo: High-fidelity neural surface reconstruction. In *CVPR*, 8456–8465.

Lin, H.; Peng, S.; Chen, J.; Peng, S.; Sun, J.; Liu, M.; Bao, H.; Feng, J.; Zhou, X.; and Kang, B. 2025. Prompting depth anything for 4k resolution accurate metric depth estimation. In *CVPR*, 17070–17080.

Lin, H.; Peng, S.; Xu, Z.; Yan, Y.; Shuai, Q.; Bao, H.; and Zhou, X. 2022. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia*, 1–9.

Lipson, L.; Teed, Z.; and Deng, J. 2021. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *3DV*, 218–227.

Liu, A.; Tucker, R.; Jampani, V.; Makadia, A.; Snavely, N.; and Kanazawa, A. 2021. Infinite nature: Perpetual view generation of natural scenes from a single image. In *ICCV*, 14458–14467.

Liu, T.; Wang, G.; Hu, S.; Shen, L.; Ye, X.; Zang, Y.; Cao, Z.; Li, W.; and Liu, Z. 2024. Fast Generalizable Gaussian Splatting Reconstruction from Multi-View Stereo. In *ECCV*, 37–53.

Lu, J.; Huang, T.; Li, P.; Dou, Z.; Lin, C.; Cui, Z.; Dong, Z.; Yeung, S.-K.; Wang, W.; and Liu, Y. 2025. Align3r: Aligned monocular depth estimation for dynamic videos. In *CVPR*, 22820–22830.

Lu, T.; Yu, M.; Xu, L.; Xiangli, Y.; Wang, L.; Lin, D.; and Dai, B. 2024. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *CVPR*, 20654–20664.

Luiten, J.; Kopanas, G.; Leibe, B.; and Ramanan, D. 2024. Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis. In *3DV*.

Lyu, X.; Sun, Y.-T.; Huang, Y.-H.; Wu, X.; Yang, Z.; Chen, Y.; Pang, J.; and Qi, X. 2024. 3dgsr: Implicit surface reconstruction with 3d gaussian splatting. *ACM TOG*, 43(6): 1–12.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 405–421.

Schonberger, J. L.; and Frahm, J.-M. 2016. Structure-from-motion revisited. In *CVPR*, 4104–4113.

Schönberger, J. L.; Zheng, E.; Frahm, J.-M.; and Pollefeys, M. 2016. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 501–518.

Seitz, S. M.; and Dyer, C. R. 1999. Photorealistic scene reconstruction by voxel coloring. *IJCV*, 35: 151–173.

Smart, B.; Zheng, C.; Laina, I.; and Prisacariu, V. A. 2024. Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. *arXiv preprint arXiv:2408.13912*.

Sun, J.; Jiao, H.; Li, G.; Zhang, Z.; Zhao, L.; and Xing, W. 2024. 3dgstream: On-the-fly training of 3d gaussians for efficient streaming of photo-realistic free-viewpoint videos. In *CVPR*, 20675–20685.

Sun, Y.-T.; Huang, Y.; Ma, L.; Lyu, X.; Cao, Y.-P.; and Qi, X. 2025. Splatter a video: Video gaussian representation for versatile processing. *NeurIPS*, 37: 50401–50425.

Teed, Z.; and Deng, J. 2020. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 402–419.

Tu, H.; Shao, R.; Dong, X.; Zheng, S.; Zhang, H.; Chen, L.; Wang, M.; Li, W.; Ma, S.; Zhang, S.; et al. 2024. TeleAloha: A Telepresence System with Low-budget and High-authenticity Using Sparse RGB Cameras. In *SIGGRAPH*, 1–12.

Wang, J.; Chen, M.; Karaev, N.; Vedaldi, A.; Rupprecht, C.; and Novotny, D. 2025a. Vggt: Visual geometry grounded transformer. In *CVPR*, 5294–5306.

Wang, P.; Liu, L.; Liu, Y.; Theobalt, C.; Komura, T.; and Wang, W. 2021a. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *NeurIPS*, 34: 27171–27183.

Wang, Q.; Wang, Z.; Genova, K.; Srinivasan, P. P.; Zhou, H.; Barron, J. T.; Martin-Brualla, R.; Snavely, N.; and Funkhouser, T. 2021b. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, 4690–4699.

Wang, R.; Xu, S.; Dai, C.; Xiang, J.; Deng, Y.; Tong, X.; and Yang, J. 2025b. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *CVPR*, 5261–5271.

Wang, S.; Leroy, V.; Cabon, Y.; Chidlovskii, B.; and Revaud, J. 2024. Dust3r: Geometric 3d vision made easy. In *CVPR*, 20697–20709.

Wang, Y.; Han, Q.; Habermann, M.; Daniilidis, K.; Theobalt, C.; and Liu, L. 2023. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *ICCV*, 3295–3306.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4): 600–612.

Wu, G.; Yi, T.; Fang, J.; Xie, L.; Zhang, X.; Wei, W.; Liu, W.; Tian, Q.; and Wang, X. 2024. 4D gaussian splatting for real-time dynamic scene rendering. In *CVPR*, 20310–20320.

Xu, H.; Peng, S.; Wang, F.; Blum, H.; Barath, D.; Geiger, A.; and Pollefeys, M. 2025. Depthsplat: Connecting gaussian splatting and depth. In *CVPR*, 16453–16363.

Xu, Q.; Xu, Z.; Philip, J.; Bi, S.; Shu, Z.; Sunkavalli, K.; and Neumann, U. 2022. Point-nerf: Point-based neural radiance fields. In *CVPR*, 5438–5448.

Xu, Z.; Peng, S.; Lin, H.; He, G.; Sun, J.; Shen, Y.; Bao, H.; and Zhou, X. 2024a. 4k4d: Real-time 4d view synthesis at 4k resolution. In *CVPR*, 20029–20040.

Xu, Z.; Xu, Y.; Yu, Z.; Peng, S.; Sun, J.; Bao, H.; and Zhou, X. 2024b. Representing long volumetric video with temporal gaussian hierarchy. *ACM TOG*, 43(6): 1–18.

Yan, J.; Wei, Z.; Yi, H.; Ding, M.; Zhang, R.; Chen, Y.; Wang, G.; and Tai, Y.-W. 2020. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *ECCV*, 674–689.

Yan, Y.; Lin, H.; Zhou, C.; Wang, W.; Sun, H.; Zhan, K.; Lang, X.; Zhou, X.; and Peng, S. 2024. Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In *ECCV*, 156–173.

Yang, J.; Mao, W.; Alvarez, J. M.; and Liu, M. 2020. Cost volume pyramid based depth inference for multi-view stereo. In *CVPR*, 4877–4886.

Yang, Z.; Gao, X.; Zhou, W.; Jiao, S.; Zhang, Y.; and Jin, X. 2024. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *CVPR*, 20331–20341.

Yao, Y.; Luo, Z.; Li, S.; Fang, T.; and Quan, L. 2018. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, 767–783.

Ye, B.; Liu, S.; Xu, H.; Li, X.; Pollefeys, M.; Yang, M.-H.; and Peng, S. 2025. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. In *ICLR*.

Yu, A.; Ye, V.; Tancik, M.; and Kanazawa, A. 2021a. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, 4578–4587.

Yu, T.; Zheng, Z.; Guo, K.; Liu, P.; Dai, Q.; and Liu, Y. 2021b. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *CVPR*, 5746–5756.

Yu, Z.; Chen, A.; Huang, B.; Sattler, T.; and Geiger, A. 2024. Mip-splatting: Alias-free 3d gaussian splatting. In *CVPR*, 19447–19456.

Zabih, R.; and Woodfill, J. 1994. Non-parametric local transforms for computing visual correspondence. In *ECCV*, 151–158.

Zhang, J.; Herrmann, C.; Hur, J.; Jampani, V.; Darrell, T.; Cole, F.; Sun, D.; and Yang, M.-H. 2025. Monst3r: A simple approach for estimating geometry in the presence of motion. In *ICLR*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 586–595.

Zheng, S.; Zhou, B.; Shao, R.; Liu, B.; Zhang, S.; Nie, L.; and Liu, Y. 2024. GPS-Gaussian: Generalizable Pixel-wise 3D Gaussian Splatting for Real-time Human Novel View Synthesis. In *CVPR*, 19680–19690.

Zhou, B.; Zheng, S.; Tu, H.; Shao, R.; Liu, B.; Zhang, S.; Nie, L.; and Liu, Y. 2025. GPS-Gaussian+: Generalizable Pixel-wise 3D Gaussian Splatting for Real-Time Human-Scene Rendering from Sparse Views. *IEEE TPAMI*.

Zhou, T.; Tucker, R.; Flynn, J.; Fyffe, G.; and Snavely, N. 2018. Stereo magnification: learning view synthesis using multiplane images. *ACM TOG*, 37(4): 1–12.