3DiffGS: 3D Gaussian Splats from Unposed Images using Ray Diffusion

Abhigyan Bhusal[1,2]
abhigyan.bhusal@naamii.org.np
Bishad Koju[2]
bishad@nhu.ai
Danda Pani Paudel[1,2,3]
danda.paudel@insait.ai
François Rameau[1,4]
francois.rameau@sunykorea.ac.kr

[1] NAAMII
Nepal
[2] NHU AI
Nepal
[3] INSAIT
Sofia University "St. Kliment Ohridski"
[4] SUNY Korea
Incheon, Korea

Abstract

Recent feed-forward 3D reconstruction methods suffer from (1) a critical supervision bottleneck of requiring expensive 3D data and (2) limited generalization. Our experiments demonstrate that diffusion models yield significantly better generalization; however, such methods exist only for camera pose estimation – not for 3D reconstruction. To enable better generalization and 3D-free training, we introduce **3DiffGS**, a camera ray diffusion model that also predicts 3D geometry. Unlike diffusion methods that focus entirely on camera estimation (e.g., PoseDiffusion and RayDiffusion) or depend on explicit 3D data (e.g., DiffusionSfM), **3DiffGS** performs diffusion directly in ray space and decodes 3D Gaussian Splats from the same ray features, naturally unifying camera and scene structure prediction. This also enabled training without 3D, using only 2D images via re-rendering. However, we observed that a trivial way of decoding Gaussians and denoised rays leads to inconsistent reconstructions. Therefore, we propose to couple camera ray and geometry predictions using a rendering-based consistency constraint. This encourages structurally coherent prediction of cameras and geometry, while also boosting camera estimation accuracy. **3DiffGS** achieves state-of-the-art sparse-view camera pose estimation results among models trained without ground-truth 3D geometry, while utilizing only *half the model size* of RayDiffusion and demonstrating superior cross-category generalization. Code available at https://github.com/abhigyan7/3DiffGS.

1 Introduction

Recent feed-forward 3D reconstruction methods, such as VGGT [33] and Fast3R [33], have demonstrated strong performance in camera pose and scene geometry estimation. However, they suffer from a critical supervision bottleneck: a reliance on strong ground-truth 3D data, typically in the form of dense point clouds or multi-view consistent depth maps. Such supervision is not only expensive and difficult to scale, but also inherently noisy, making it a critical bottleneck for learning robust and generalizable 3D representations.

© 2026. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

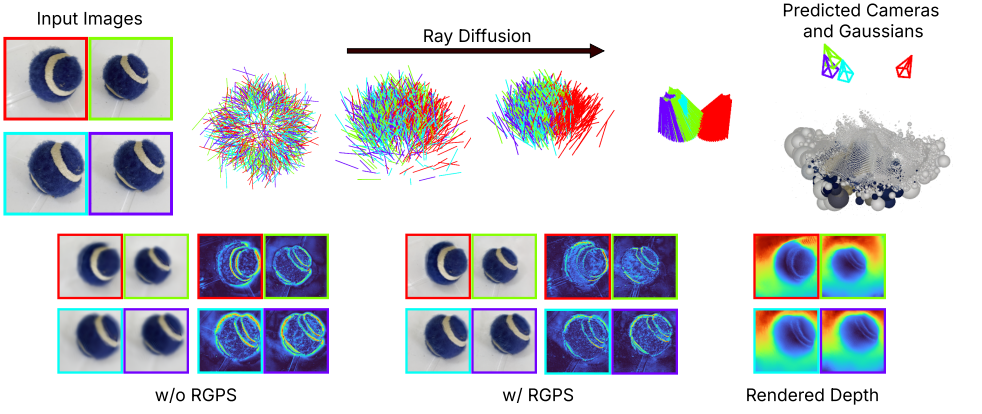


Figure 1: **Joint camera and geometry estimation with 3DiffGS.** *Top:* Our model denoises camera rays and regresses 3D Gaussians given unposed images. *Bottom:* Unconstrained joint prediction leads to camera-Gaussian divergence, resulting in high photometric error in renders from predicted cameras (left). Minimizing photometric error from predicted cameras during training helps alignment and pose accuracy (middle). Rendered depth maps (right).

Beyond this supervision issue, a more fundamental challenge arises from the nature of joint 3D perception itself. Camera pose estimation and scene geometry estimation are tightly coupled problems: accurate geometry depends on correct viewpoints, while accurate viewpoints depend on reliable geometry. However, most existing feed-forward approaches decouple these two predictions into independent branches. This factorization simplifies learning, but breaks the intrinsic geometric consistency of the problem, often resulting in subtle but compounding inconsistencies between estimated poses and reconstructed structure. While post-hoc optimization techniques such as Bundle Adjustment [33] or photometric refinement [48] can partially correct these errors, they fundamentally rely on strong initialization and iterative refinement, which contradicts the goal of fast feed-forward inference. This tension highlights a deeper difficulty: learning a model that can simultaneously reason about camera geometry and 3D structure in a consistent way, under weak supervision and without iterative correction, is inherently non-trivial. It requires not only a representation that can unify pose and geometry, but also a learning mechanism that enforces their consistency during training instead of post-hoc.

These challenges are further compounded in the sparse-view, wide-baseline setting. Here, limited visual overlap and object symmetries induce multi-modal viewpoint ambiguity [44, 45, 49]. When faced with such uncertainty, deterministic regressors must commit to a single point estimate, which might be inaccurate [45]. Generative modeling provides a natural solution in such ambiguous scenarios. Empirical evidence supports this: diffusion based methods (PoseDiffusion [32] and RayDiffusion [45]) show that diffusion outperforms regression for camera pose estimation due to the mentioned reason. In addition, our controlled studies confirm significantly better zero-shot cross-category generalization for camera poses across pose-only, photometric, and geometric supervision regimes (See Section 4.3). Yet, a critical gap remains: existing diffusion models either lack explicit 3D structure prediction [32, 45]

or still rely on ground-truth 3D supervision [49]. In this context, a diffusion-based method that learns without 3D supervision, yet natively predicts explicit geometry, does not exist in the literature.

Motivated by this, we propose **3DiffGS**, a diffusion-based framework for joint 3D reconstruction and camera estimation from sparse, unposed images (Figure 1). The core idea is to perform diffusion directly in ray space, and decode 3D Gaussian Splats from the same features used to denoise rays. This design provides a compact and scalable representation that naturally predicts camera via diffusion and regresses scene structure from the same representation, while avoiding explicit 3D supervision through the use of 2D rendering-based learning.

However, naively decoding geometry from ray features risks inconsistent reconstructions, as small deviations propagate into large geometric error. To resolve this, we introduce a rendering-guided training scheme enforcing photometric consistency between predicted rays and the reconstructed Gaussian scene. Unlike post-hoc correction methods, this mechanism couples diffusion-based ray prediction and geometric reconstruction during training, boosting consistency between rays and Gaussians, while concurrently improving camera pose accuracy.

As a result, **3DiffGS** achieves state-of-the-art performance on the sparse-view CO3Dv2 camera pose benchmark among models trained without geometry supervision, using only half the model size of RayDiffusion. Furthermore, it generalizes better than a diffusion-free feed-forward baseline in both camera pose and geometry estimation, validating ray-space diffusion as a highly effective representation for unposed 3D reconstruction.

In summary, our contributions are:

1. A ray-space diffusion based formulation for joint camera and 3D Gaussian Splat reconstruction under weak supervision of only 2D images and their poses.
2. A rendering-guided training strategy that enforces photometric consistency between predicted rays and reconstructed scenes, and also improves camera pose estimation.
3. Exhaustive experimental evaluations on 41 seen and 10 unseen categories of CO3Dv2, consisting of 20K+ scenes and 2.2M images in the training set and 2.5K+ test scenes, where our method offers a new SOTA camera pose estimation performance among methods trained without ground truth geometry.

2 Related Work

Classical and learned structure-from-motion. Traditional Structure-from-Motion (SfM) pipelines, such as COLMAP [27], estimate camera poses and 3D structure through a two-stage process involving feature matching and global optimization. While recent learning-based improvements, including learned feature extractors [5, 6, 24], matchers [17, 26, 29], and bundle adjustment [16, 30] improve robustness, they remain fundamentally constrained by their staged design, where reconstruction quality is tightly coupled to the success of intermediate matching steps. More recent feed-forward alternatives, such as VGGT [33] and Fast3R [38], replace this pipeline with direct regression of poses and scene structure, achieving strong performance in multi-view reconstruction. However, these methods rely heavily on strong geometric supervision (e.g., depth or point clouds) and typically decouple pose and geometry prediction, often requiring post-hoc refinement such as Bundle Adjustment to ensure consistency.

Table 1: **Comparison of data-driven 3D vision methods.** Our method utilizes a diffusion framework to estimate camera parameters, with photometric supervision to produce re-renderable Gaussian Splatting scenes from sparse views without requiring 3D geometry supervision.

	Type		Supervision			Output		
Method	Diff.	Regr.	Pose	3D	Image	Pose	Scene Representation	Renderable
RayRegression [45]		✓	✓			✓	None	
VGGT [53]		✓	✓	✓		✓	Point Cloud	
AnySplat [11]		✓	✓	✓	✓	✓	3D Gaussian Splat	✓
YoNoSplat [41]		✓	✓		✓	✓	3D Gaussian Splat	✓
E-RayZer [50]		✓			✓	✓	3D Gaussian Splat	✓
RayDiffusion [45]	✓		✓			✓	None	
DiffusionSfm [49]	✓		✓	✓		✓	Point Cloud	
Ours	✓		✓		✓	✓	**3D Gaussian Splat**	✓

Feed-forward 3D reconstruction with neural representations. Neural scene representations, including NeRF [18], 3D Gaussian Splatting [12], and their variants [1, 2, 7, 19, 39, 42], enable high-quality novel view synthesis but require per-scene optimization and accurate camera poses. To improve efficiency, feed-forward approaches [3, 4, 9, 37, 43, 46] amortize this optimization by learning geometric priors from large-scale datasets. However, most methods still assume known camera poses and are limited to few input views. Pose-free methods [40, 48] remove this requirement by directly predicting scene representations (e.g., radiance fields or Gaussians) from unposed images, but rely on regression-based objectives, require known intrinsics at inference, and are typically designed for constrained scene settings. Recent method YoNoSplat[41] and AnySplat [11] predict camera parameters and 3D Gaussian Splats jointly, but still require weights initialization from pretrained 3D Vision networks [53, 34].

Diffusion-based camera estimation and 3D reconstruction. Diffusion models [8, 21] have emerged as a powerful alternative for geometric reasoning under uncertainty. Methods such as PoseDiffusion [32] and RayDiffusion [45] formulate camera estimation as a denoising process over pose or ray representations, achieving strong robustness in sparse-view settings. However, these approaches do not reconstruct explicit scene geometry. DiffusionSfM [49] extends this idea by jointly modeling camera rays and 3D structure, but relies on explicit geometric supervision . In contrast, our method performs diffusion directly in ray space and decodes 3DGS from predicted ray features, enabling joint pose and geometry estimation under weak supervision using only photometric rendering.

Self-supervised and prior-free reconstruction. A related line of work explores self-supervised 3D reconstruction from unposed images using neural rendering. Early methods [13, 31, 36, 51], predict scene features from a single image and supervise the renders from target views. Methods such as RayZer [10] and RUST [25] leverage neural rendering and photometric consistency for view synthesis. Concurrent work E-RayZer [50] incorporates Gaussian Splatting for improved 3D grounding. While these approaches move toward supervision-free learning, they target general multi-view reconstruction, require a training curriculum over dense views in the case of RayZer and E-Rayzer, and follow a pose-then-scene paradigm during prediction. In contrast, our focus is on sparse-view, object-centric reconstruction under wide-baseline settings, and joint prediction of camera and scene using

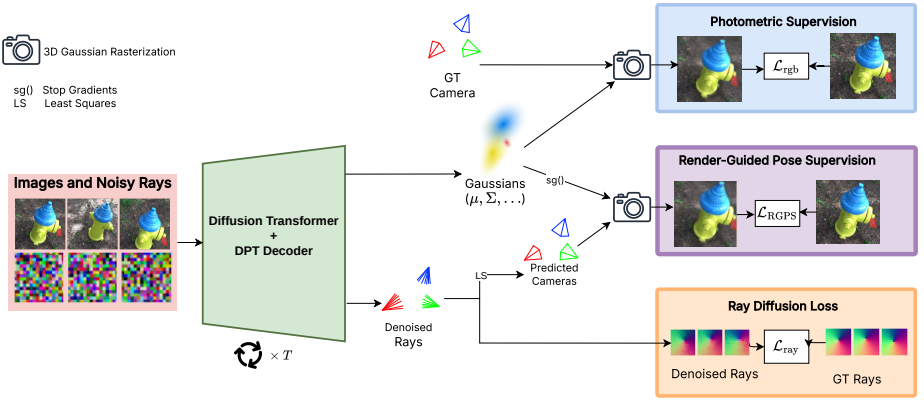


Figure 2: **Overview of 3DiffGS.** Given a set of sparse input images, our method predicts patch-wise rays and pixel-wise Gaussian parameters using a forward diffusion process. A Diffusion Transformer denoises camera rays conditioned on the context images, and a Dense Prediction Transformer head decodes 3D Gaussians from the intermediate features of the Diffusion Transformer. The model is trained with a diffusion objective on the denoised rays, photometric objectives on the images rendered using ground truth cameras, and Render-Guided Pose Supervision (RGPS) using predicted cameras.

a unified architecture.

Summary. Overall, existing methods either rely on strong geometric supervision, decouple pose and geometry prediction, or are regression-based (Table 1). Our work bridges these gaps by combining diffusion in ray space with Gaussian Splat decoding under rendering-based supervision, enabling a unified and scalable formulation for joint pose estimation and 3D reconstruction.

3 Method

Given a sparse set of unposed images, 3DiffGS jointly estimates the underlying 3D scene geometry and camera parameters (see Figure 2). We begin by defining our dual representations: modeling the scene via dense, pixel-wise 3D Gaussians and the cameras via a regular grid of rays (Section 3.1). We then detail our denoising diffusion transformer architecture with auxiliary Gaussian prediction and training losses in Section 3.2. To prevent our rays and Gaussian predictions from drifting apart, we introduce Render-Guided Pose Supervision (RGPS) in Section 3.3, which enforces photometric consistency between predicted cameras and Gaussians.

3.1 3D Scene Representation, Notation, and Coordinate Frame

Given a set of N input images of resolution $H \times W$ with corresponding intrinsics $\mathbf{K}_i$ and extrinsics $\mathbf{T}_i$, we associate each patch p with a ray originating from the camera center $\mathbf{o}_{i,p}$ and unit direction $\mathbf{d}_{i,p}$ pointing towards the patch center on the image sensor. These components

are mapped into Plücker coordinates to form the ray representation $\mathbf{x}_{i,p} = \mathbf{d}_{i,p} \oplus \mathbf{o}_{i,p} \times \mathbf{d}_{i,p}$. Pinhole camera parameters can then be recovered from the grid of camera rays using a least squares process [45, 49], described in Appendix E. Crucially, this rays to camera parameters mapping is differentiable, which enables Render-Guided Pose Supervision (RGPS) (Section 3.3).

In parallel, we associate each pixel $\mathbf{u}$ in image i with a set of 3D Gaussian parameters $\Theta_{i,\mathbf{u}} = \{\mu, \alpha, \mathbf{q}, \mathbf{s}, \mathbf{c}\}_{i,\mathbf{u}}$ where $\mu_{i,\mathbf{u}} \in \mathbb{R}^3$ is the Gaussian center, $\alpha_{i,\mathbf{u}} \in \mathbb{R}^+$ denotes the opacity, $\mathbf{q}_{i,\mathbf{u}} \in \mathbb{H}$ is the rotation parameterized as a quaternion, $\mathbf{s}_{i,\mathbf{u}} \in \mathbb{R}^3_+$ is the scale, and $\mathbf{c}_{i,\mathbf{u}} \in \mathbb{R}^k$ represents spherical harmonic coefficients.

In our representation, all Gaussians and camera rays exist in a single canonical frame, defined by the origin at the least squares intersection of all camera axes, the first camera having identity rotation and unit distance away from the origin, following RayDiffusion [45].

3.2 3DiffGS: 3D Diffusion for Gaussian Splatting

3.2.1 Modeling

Since we assume we have the ground truth values for only the camera parameters, and not for scene geometry, we adopt a hybrid modeling approach where we define a diffusion process over the ray representations $\mathbf{x}_{i,p}$, and treat the Gaussian parameters $\Theta_{i,\mathbf{u}}$ as auxiliary deterministic predictions at each timestep t.

Let $\mathbf{x}^t_{i,p}$ denote the noisy ray representation at time step t. Our model learns a denoising function $f_\theta(\mathbf{x}^t_{i,p}, t|I)$ to jointly predict both the clean rays and 3D Gaussian Splats conditioned on the input images. Crucially, the gaussian parameters Θ do not undergo diffusion. Instead, at each timestep t, the model predicts a set of gaussian parameters along with the current estimate of the denoised rays.

3.2.2 Architecture

We implement our model as a denoising diffusion transformer (DiT) [21] coupled with a dense prediction head. For each input image, we extract $P \times P$ patch-wise features using a pretrained DINOv2 [20] encoder, which we concatenate with patch-wise noisy ray representation. We then augment these patch features with patch and view positional encodings, and process the resulting patch tokens with a Diffusion Transformer [21]. The DiT jointly processes all $N \times P \times P$ tokens and outputs denoised camera rays. We then tap into the intermediate activations in the DiT at 4 different levels, and process them using a Dense Prediction Transformer (DPT) [22], which outputs pixel-aligned Gaussians for each of the input image pixel. Because the Gaussians are all part of the same canonical coordinate frame as described in Section 3.1, we simply concatenate the Gaussians from all views before rasterization.

3.2.3 Training Objectives

To optimize the 3DiffGS model, we use the following losses:

- **Camera Ray Diffusion ($\mathcal{L}_{\text{ray}}$):** We employ an x_0-prediction objective and minimize the ℓ_2 loss between the predicted denoised rays $\hat{\mathbf{x}}^0_{i,p}$ and the ground truth Plücker coordinates $\mathbf{x}_{i,p}$ across all diffusion timesteps.

- **Photometric Reconstruction ($\mathcal{L}_{\text{rgb}}$, $\mathcal{L}_{\text{lpips}}$):** We minimize the ℓ_2 photometric error and a VGG [28]-based perceptual loss [47] between the input images I_i and the rasters $\hat{I}_i$ rendered using ground-truth poses.
- **Novel-View Consistency ($\mathcal{L}_{\text{nv}}$):** To ensure the 3D Gaussians do not overfit to the context views, we supervise renders from novel viewpoints I_j sampled within a fixed frame index offset from the context views. This loss is weighted by the rendered alpha mask to focus supervision on geometry observed only in the context views.
- **Mean Regularization ($\mathcal{L}_{\text{reg}}$):** Gaussians that lie outside the camera frustum receive no photometric gradients during training. To encourage the predicted Gaussians to be visible through context viewpoints and well-distributed across the view, we apply a cosine similarity penalty between the ray passing through the ground-truth camera center and a Gaussian mean, and the ray passing through the ground-truth camera center and the corresponding pixel center.
- **Regularization:** We further apply an image-gradient weighted total variation (TV) loss ($\mathcal{L}_{\text{tv}}$) on the rendered depth maps to promote smoothness, and regularize Gaussian scales larger than a threshold γ_{scale} in order to prevent Out-of-Memory errors during training: $\mathcal{L}_{\text{scale}} = \frac{1}{NHW} \sum_{i,u} \max(0, s_{i,u} - \gamma_{\text{scale}})^2$.

3.3 Enforcing consistency using Render-Guided Pose Supervision (RGPS)

Because our architecture decouples the camera ray diffusion from the 3D Gaussian predictions, the two predictions are not inherently coupled. Standard photometric supervision from ground-truth camera extrinsics and intrinsics leaves the predicted rays constrained just by the ground truth camera rays during training, and not coupled with the predicted scene geometry.

To explicitly enforce consistency between these outputs, we implement Render-Guided Pose Supervision (RGPS) in which we optimize an additional rendering loss using our *predicted camera parameters*. Let $\hat{\pi}_i$ represent the camera extrinsics and intrinsics recovered from the denoised ray estimates $\hat{x}_i$. We apply the differentiable splatting operation to the predicted Gaussians $\hat{\Theta}_{i,\mathbf{u}}$ using $\hat{\pi}_i$, and penalize the ℓ_2 loss with ground truth images I_i. We allow gradients to flow only through the predicted rays and not through the predicted Gaussians, which helps training stability.

4 Experiments

Following our setup in Section 4.1, we structure our experiments into four parts: CO3D sparse-view pose and geometry evaluation (Section 4.2), controlled diffusion vs. regression analysis (Section 4.3), ablation studies for RGPS and loss terms (Section 4.4), and unposed novel view synthesis (Section 4.5).

4.1 Experimental Setup

4.1.1 Dataset and Evaluation Protocol

We train and evaluate our models on CO3Dv2 [23], which is an object-level dataset consisting of images, ground truth depths, and poses of objects across 51 common categories (details in Appendix B).

Full CO3Dv2 benchmark: We train on 41 categories and evaluate both on these 41 *seen* categories and 10 *unseen* categories, following previous works [15, 45, 49].
Single-Category Experiments: To isolate the architectural impact of diffusion versus regression, we evaluate both paradigms independently under three distinct supervision regimes (pose-only, photometric, and geometric). We train these variants exclusively on the *hydrant* category of CO3Dv2 (600 training and 50 test scenes) to assess both in-domain capacity and out-of-distribution generalization to a set of 5 held-out categories, namely *bench, car, chair, teddybear, vase*.
Novel View Synthesis: We evaluate the novel-view synthesis performance on unseen CO3D categories with 2, 4, and 6 input context views. We first predict pose and gaussians for two sets of views: context only, and context+target. Then, we align the predicted camera centers in the latter set with the former, and use the resulting aligned target poses to render the scene predicted with context only input. This evaluation protocol is adapted from AnySplat [11], but we do full Sim(3) alignment instead of just scale, which would be unfair to models whose origin could change with addition of views. We evaluate with {2, 4, 6} context views and 2 target views.

In all experiments, we use the same test view orderings as used by previous sparse-view camera pose estimation methods [15, 45, 49].

4.1.2 Baselines and Model Variants

For the full CO3Dv2 sparse-view benchmark, we evaluate 3DiffGS against our primary baselines RayDiffusion [45] and a joint camera ray and 3D Gaussian Splatting regression baseline. We also include VGGT*† [33], AnySplat*† [11], and DiffusionSfM [49], evaluating both CO3D-only (†) and mixture-trained variants (*†) of the latter. In this context, * denotes training on large-scale dataset mixtures (e.g., those used in Mast3r [24]), while † indicates the use of ground-truth 3D geometry data during training. RayDiffusion, DiffusionSfM, VGGT, and Anysplat have 387 million, 408 million, 1.26 billion, and 1.19 billion trainable parameters respectively, while 3DiffGS and our regression baseline both have 179 million trainable parameters each.

In the controlled single-category study, we evaluate regression and diffusion counterparts across three distinct regimes: pose-only supervision, joint pose and geometric supervision, and joint pose and photometric supervision. To prevent overfitting on this single-category setup, we proportionally scale down the parameter capacity across all evaluated variants and train each to convergence. The regression variants for all models were created by zeroing out the timestep and noisy rays inputs, so as to keep the architecture and training pipeline the same. We implement the geometry-supervised models as ray origin and endpoint predictors using DiffusionSfM's training pipeline [49].

All models process input images at a 224×224 resolution, except VGGT [33] and AnySplat [11] whose inputs are resized to 518×518 and 448×448 pixels respectively in line with their training. For computational and memory efficiency during training, our Gaussian parameter maps are predicted at a 112×112 spatial resolution. We provide architectural details in Appendix C and training details and hyper-parameters in Appendix D.

4.1.3 Metrics

Camera Pose Accuracy: Following prior work [15, 45, 49], we report two metrics to evaluate camera pose: **(1) Camera Rotation Accuracy**, defined as the proportion of predicted

relative rotations within 15° of the ground truth, and **(2) Camera Center Accuracy**, defined as the proportion of predicted camera positions that lie within 10% of the scene extent of the ground truth, after an optimal Sim(3) alignment (making Camera Center Accuracy 100% for N=2 images).

Photometric Consistency: To measure the alignment between our predicted cameras and 3D Gaussians, we evaluate the Peak Signal-to-Noise Ratio (PSNR) between the input images and their corresponding renders from the predicted cameras. Unlike Novel View Synthesis metrics that evaluate generalization to holdout views, this self-reconstruction PSNR explicitly measure internal consistency, higher values demonstrating that the jointly predicted geometry and poses accurately explain the observed inputs.

Geometry Quality: To evaluate the accuracy of the predicted geometry, we report the Chamfer Distance for both the complete set of scene points (CD) and exclusively for the foreground object points (CD_{obj}).

Novel View Synthesis Quality: To evaluate the novel view synthesis quality of the predicted 3D Gaussians, we report the standard PSNR, perceptual similarity [47], and structural similarity[35] between novel view renders and ground truth novel view images.

4.2 CO3Dv2 Sparse View Benchmark

Table 2: **Evaluation on CO3D:** We report camera centers (CC) within 10% of scene scale, relative rotations (R) within 15°, photometric consistency PSNR, and Chamfer Distance for all (CD) and foreground (CD_{obj}) points. Higher is better (↑) for CC, R, and PSNR; lower is better (↓) for CD metrics. Metrics are averaged across 2–8 input views. Best results among models trained without 3D supervision are in **bold**. *Trained on additional datasets. †Trained with ground-truth 3D supervision.

Method	Seen			Unseen				
	CC	R@15°	PSNR	CC	R@15°	PSNR	CD	CD_{obj}
VGGT*† [33]	96.7	97.8	–	92.8	97.2	–	0.029	0.016
AnySplat*† [11]	96.2	97.6	23.6	92.3	97.1	24.4	0.050	0.022
DiffusionSfM*† [49]	93.6	94.7	–	89.2	94.0	–	0.025	0.023
DiffusionSfM† [49]	93.4	94.6	–	87.5	92.5	–	0.021	0.024
RayDiffusion [45]	**89.7**	**92.8**	–	80.5	86.4	–	–	–
Regression Baseline	81.2	84.1	20.6	65.7	70.3	20.7	0.187	0.084
3DiffGS (Ours)	**89.7**	92.3	**22.5**	**82.2**	**88.3**	**23.2**	**0.109**	**0.052**

We report the pose prediction accuracy across both seen and unseen categories in Table 2, and visualize qualitative results on scenes from the unseen categories of the CO3Dv2 benchmark in Figure 3.

Quantitative Evaluation: As shown in Table 2, 3DiffGS achieves state-of-the-art performance among methods trained without 3D scene geometry supervision. Our method outperforms our closest pose-supervised counterpart RayDiffusion [45] in the *unseen* categories, achieving a camera center accuracy (CC) of **82.2%** (compared to RayDiffusion's 80.5%) and a rotation accuracy (R@15°) of **88.3%** (compared to 86.4%). This performance gap demonstrates that jointly modeling 3D scene structure alongside camera rays and enforcing photometric consistency provides a mutually beneficial optimization signal, resulting in bet-

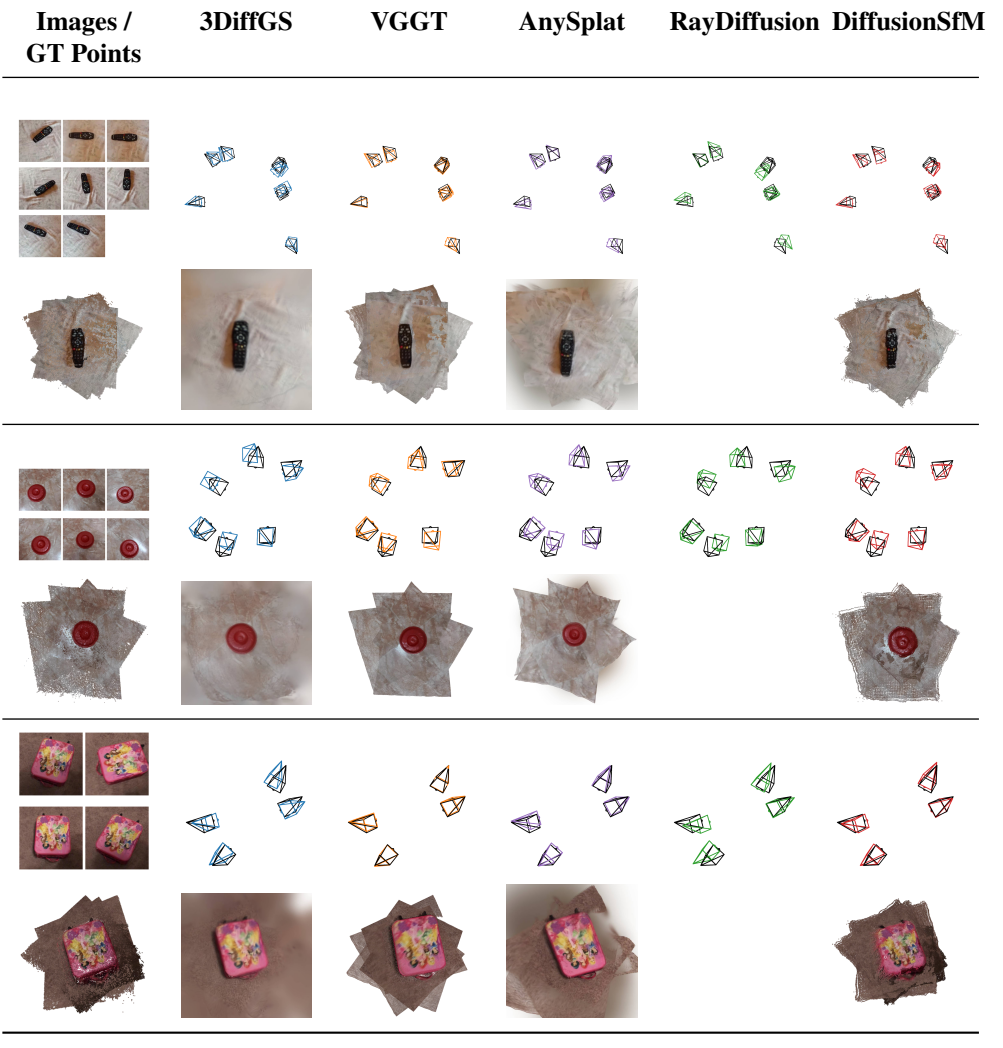


Figure 3: **Qualitative comparison:** Predicted camera poses and point clouds/RGB renderings on *unseen* categories of CO3Dv2. Top: camera poses; bottom: point clouds for Ground Truth, VGGT, and DiffusionSFM, and RGB renderings for AnySplat and Ours. Ground-truth camera frustums are colored black.

ter pose prediction performance compared to modeling camera rays in isolation. Compared against the joint rays-Gaussian regression baseline, our method yields a +16.5% absolute improvement in camera center accuracy and a +18.0% jump in rotation accuracy, while also decisively outperforming the regression baseline in photometric consistency and geometry quality. The gap demonstrates the effectiveness of our diffusion modelling approach and RGPS. VGGT [33], DiffusionSfM [49], and AnySplat [11] represent an upper performance bound due to their reliance on ground-truth 3D geometry and/or multi-dataset mixture training. We include the pose accuracy metrics for each number of views in the full pose evaluation table in Appendix A, and view-wise chamfer distance breakdown for all methods and category specific breakdown for 3DiffGS in Appendix G.

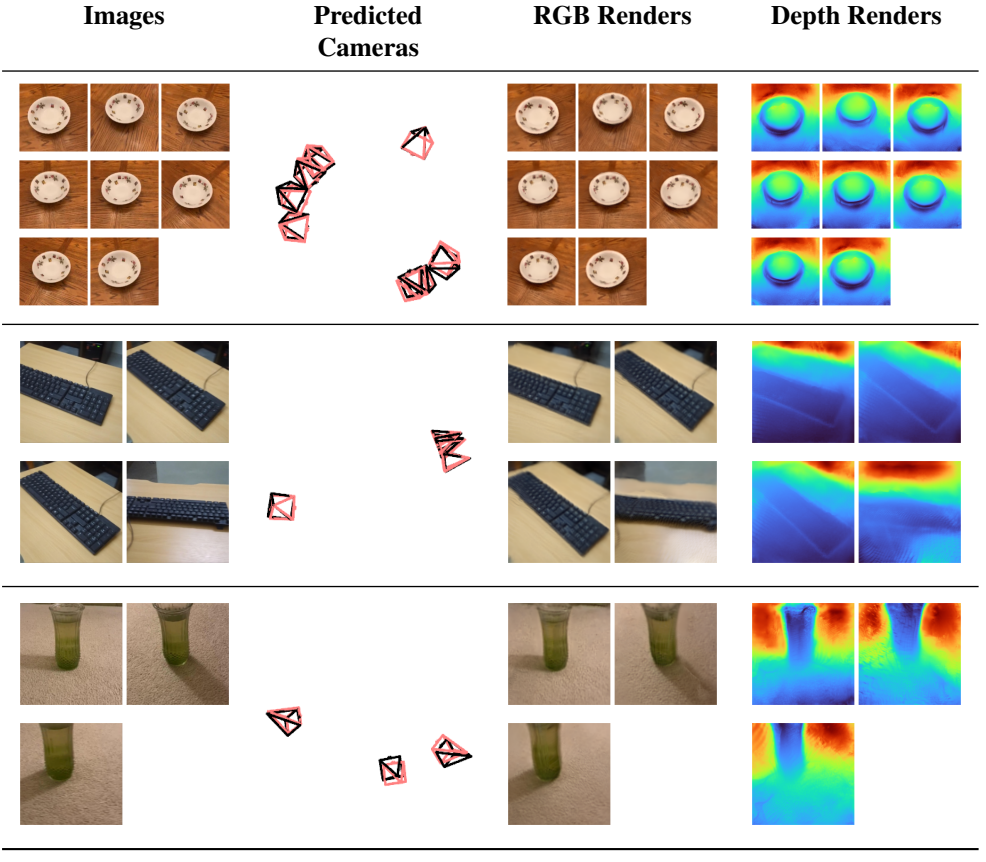


Figure 4: **Additional 3DiffGS Qualitative Results:** Predicted Cameras and Renders from Predicted Cameras. Black camera frustums are ground truths, orange frustums are predicted. 3DiffGS predicts accurate camera poses in wide-baseline images.

Qualitative Evaluation: We visualize qualitative results on unseen categories of the CO3Dv2 benchmark in Figure 3. 3DiffGS produces cohesive, well-aligned joint predictions of camera poses and 3D Gaussian Splats. Our method shows better generalization to unseen categories than RayDiffusion, and also outputs geometry, which RayDiffusion does not do. Additional qualitative camera pose estimation and renders by our method are provided in Figure 4.

4.3 Controlled Study on Diffusion Modeling

Table 3: **Generalization Evaluation of Models Trained on Hydrants:** We report the proportion in percentage of camera centers (CC) within 10% of the scene scale and relative rotations (Rot.) within given thresholds of the ground truth poses. On Unseen categories, relative improvements of diffusion models over their regression counterparts are denoted in subscripts. *Note: Geometry supervised models were trained at* 16×16 *resolution.*

Supervision			Diff.	Seen (Hydrants)		Unseen	
Pose	Geom.	Photo.		CC	R@15°	CC	R@15°
✓				92.2	93.6	29.3	18.1
✓			✓	92.5	93.0	32.2 $_{+10\%}$	24.4 $_{+35\%}$
✓	✓			81.7	78.5	27.3	15.0
✓	✓		✓	83.8	77.4	31.9 $_{+17\%}$	19.1 $_{+27\%}$
✓		✓		91.2	89.7	28.7	19.4
✓		✓	✓	92.7	94.0	33.5 $_{+17\%}$	24.3 $_{+25\%}$

We isolate the impact of diffusion modeling versus regression by training exclusively on the *hydrant* category and evaluating zero-shot generalization across unseen classes (Table 3).

On the seen category, both paradigms perform comparably, demonstrating that deterministic regression networks possess sufficient capacity for in-domain data. However, on unseen categories, the diffusion variants demonstrate much more robust predictive capabilities, yielding substantial relative gains (e.g., +35% in R@15° under pose-only supervision). This persistent margin shows that modeling the camera distribution via diffusion results in better generalization in camera prediction to out-of-distribution categories.

We note that the geometry-supervised variants were trained at a significantly lower ray resolution (16×16) than DiffusionSfM (256×256). Although we explored both DiffusionSfM's progressive low-to-high resolution training strategy and high-resolution training from scratch, these configurations yielded inferior performance to our 16×16 configuration. Further optimization of the training pipeline and hyperparameters may improve absolute performance for these variants.

4.4 Ablation Studies

We conduct ablation experiments on to validate our loss formulations and Render-Guided Pose Supervision (RGPS) on the hydrants category of CO3D.

Table 4: **Ablation Study.** Our full formulation in gray.

Variant	CC@0.1	R@15°	PSNR	CD	CD_{obj}
Diffusion Baseline	90.2	89.2	19.0	0.119	0.182
w/o $\mathcal{L}_{reg}$ (Mean Reg.)	89.1	89.3	18.0	0.152	0.362
w/o $\mathcal{L}_{nv}$ (Novel View)	87.2	83.4	17.8	0.197	0.432
w/ RGPS (Ours)	**93.1**	**90.9**	**20.2**	**0.096**	0.178
w/ RGPS, No-detach	90.0	87.6	18.6	0.111	**0.168**

We evaluate the impact of the Gaussian Mean Ray regularization loss ($\mathcal{L}_{\text{reg}}$), Novel View photometric loss ($\mathcal{L}_{\text{nv}}$), and RGPS in Table 4 and Figure 5. Removing $\mathcal{L}_{\text{reg}}$ primarily de-

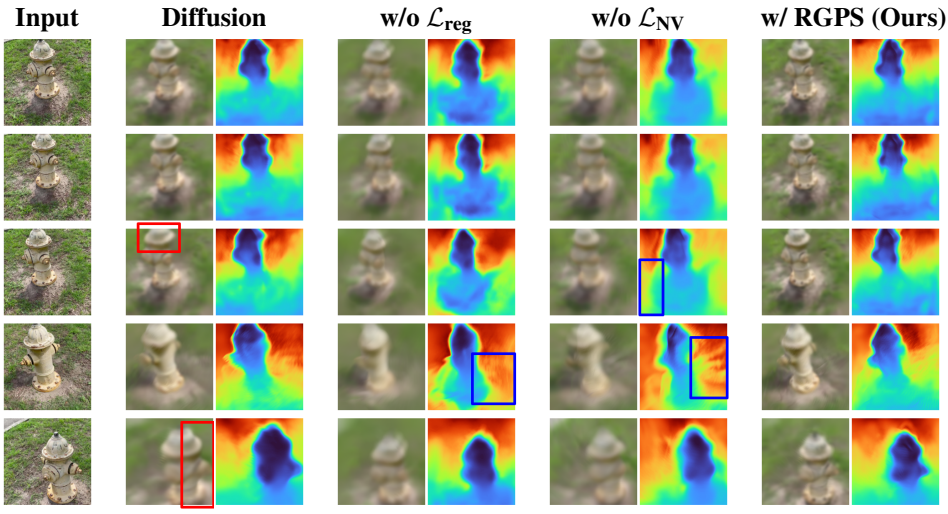


Figure 5: **Qualitative Ablation**. Compared to the diffusion baseline, removing our mean regularization loss ($\mathcal{L}_{\text{reg}}$) results in vacant regions in the depth map and blurred RGB renders. Excluding the novel-view consistency loss ($\mathcal{L}_{\text{nv}}$) leads to more severe background depth degeneracy. But the diffusion baseline still results in photometric inconsistency (see highlighted red boxes), whereas 3DiffGS (Ours) demonstrates improved photometric consistency in these regions.

grades geometry, causing the foreground Chamfer Distance (CD_{obj}) to double from 0.182 to 0.362. Visually, this manifests as poor Gaussian placement, resulting in loss of detail in the RGB renders and blank spaces in the background in the depth renders.

Removing $\mathcal{L}_{\text{nv}}$ causes degradation across both pose and geometry metrics, dropping camera rotation accuracy by nearly 6% absolute and increasing CD_{obj} to 0.432. Without novel-view supervision, the model exploits context-view shortcuts, resulting in floaters and poor geometry, especially in the background. Jointly optimizing both losses is therefore essential for multi-view consistent scene representations.

Integrating RGPS yields a +2.9% absolute increase in camera center accuracy (CC) and a +1.7% boost in rotation accuracy (R@15°), alongside a 1.2 dB gain in photometric consistency PSNR. Omitting RGPS allows the predicted cameras and the 3D Gaussians to decouple, which results in the renders from predicted cameras being out of alignment with the input images. Enabling RGPS forces a photometric constraint that anchors the joint prediction, resulting in photometric consistency along with substantially improved camera pose estimation. When the Gaussian branch is not detached, the pose accuracy and photometric consistency both degrade compared to the diffusion baseline, resulting from unstable optimization. We include the loss curves for both detached and undetached training in Appendix F.

4.5 Novel View Synthesis

As shown in Table 5, our method markedly outperforms the regression baseline across all sparse-view settings (e.g., +3.4 dB PSNR at 2 views) and yields competitive rendering quality close to AnySplat.

Table 5: NVS performance comparison across sparse views.

	2 Views			4 Views			6 Views		
Method	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM
AnySplat	**24.2**	**.245**	**.795**	**24.0**	**.252**	**.787**	**23.7**	**.262**	**.775**
Regression	20.4	.588	.624	20.2	.590	.620	19.9	.593	.617
Ours	23.8	.389	.703	23.3	.402	.691	22.8	.412	.682

5 Discussion

We presented 3DiffGS, a novel framework that bridges the gap between generative camera pose estimation and feed-forward 3D scene reconstruction. By unifying a ray-based diffusion formulation with rendering-guided photometric consistency, our approach predicts geometry in the form of 3D Gaussian Splats, without using expensive ground-truth 3D geometry during training. Furthermore, in line with the trends observed in our zero-shot experiments across three distinct supervision regimes, this diffusion-centric design achieves significantly better cross-category generalization than a standard deterministic regression baseline. Ultimately, 3DiffGS establishes a new state-of-the-art for sparse-view camera estimation among models trained without explicit 3D geometry supervision. It narrows the performance gap to fully geometry-supervised methods like DiffusionSfM, all while operating at half the parameter capacity of the prior leading baseline.

Limitations: While 3DiffGS successfully bridges the gap between diffusion based camera estimation and 3D scene reconstruction without ground-truth geometry data, several limitations remain. First, camera pose accuracy and geometric quality gaps persist compared to geometry-supervised methods like DiffusionSfM. This indicates that explicit 3D supervision still provides a stronger supervision for pose estimation than 2D photometric supervision and photometric consistency alone. Second, relying strictly on photometric rendering losses can lead to overly smooth depth boundaries and sub-optimal background geometry allocation. Extreme in-the-wild scenarios, such as scenes with massive viewpoint baselines or highly refractive materials (e.g., glassware), remain challenging to reconstruct accurately under our current weakly supervised formulation. Finally, our current object-centric experimental setup does not cover larger, unbounded, indoor and outdoor scenes and less object-centric captures.

Future Work: Future research could close the geometry gap by integrating zero-shot monocular depth or normal priors to sharpen depth boundaries without requiring expensive 3D ground-truth scans. Additionally, exploring view-dependent material formulations within the diffusion process could extend the robustness of 3DiffGS to scenes with specular or refractive components. Finally, scaling this unified formulation into a general-purpose reconstruction framework capable of seamlessly handling arbitrary view counts, ranging from sparse inputs to dense captures with thousands of images, across diverse indoor and outdoor environments represents an exciting direction toward a universal scene-level diffusion based 3D vision model.

References

[1] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. ICCV, 2021.

[2] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. CVPR, 2022.

[3] David Charatan, Sizhe Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In CVPR, 2024.

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

[5] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 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.

[6] 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.

[7] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In CVPR, 2022.

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, Advances in Neural Information Processing Systems, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

[9] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. LRM: Large reconstruction model for single image to 3d. In The Twelfth International Conference on Learning Representations, 2024. URL https://openreview.net/forum?id=sllU8vvsFF.

[10] Hanwen Jiang, Hao Tan, Peng Wang, Haian Jin, Yue Zhao, Sai Bi, Kai Zhang, Fujun Luan, Kalyan Sunkavalli, Qixing Huang, et al. Rayzer: A self-supervised large view synthesis model. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4918–4929, 2025.

[11] Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. ACM Transactions on Graphics (TOG), 44(6): 1–16, 2025.

[12] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics, 42(4), July 2023. URL https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

[13] Zihang Lai, Sifei Liu, Alexei A Efros, and Xiaolong Wang. Video autoencoder: self-supervised disentanglement of 3d structure and motion. In ICCV, 2021.

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

[15] 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), 2024.

[16] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-Perfect Structure-from-Motion with Featuremetric Refinement. In ICCV, 2021.

[17] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. LightGlue: Local Feature Matching at Light Speed. In ICCV, 2023.

[18] 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. In ECCV, 2020.

[19] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. ACM Trans. Graph., 41(4): 102:1–102:15, July 2022. doi: 10.1145/3528223.3530127. URL https://doi.org/10.1145/3528223.3530127.

[20] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[21] William Peebles and Saining Xie. Scalable diffusion models with transformers. arXiv preprint arXiv:2212.09748, 2022.

[22] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. ArXiv preprint, 2021.

[23] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In International Conference on Computer Vision, 2021.

[24] Jerome Revaud, Philippe Weinzaepfel, César Roberto de Souza, and Martin Humenberger. R2D2: repeatable and reliable detector and descriptor. In NeurIPS, 2019.

[25] Mehdi S. M. Sajjadi, Aravindh Mahendran, Thomas Kipf, Etienne Pot, Daniel Duckworth, Mario Lučić, and Klaus Greff. RUST: Latent Neural Scene Representations from Unposed Imagery. CVPR, 2023.

[26] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In CVPR, 2020. URL https://arxiv.org/abs/1911.11763.

[27] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In International Conference on Learning Representations, 2015.

[29] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. CVPR, 2021.

[30] Chengzhou Tang and Ping Tan. BA-net: Dense bundle adjustment networks. In International Conference on Learning Representations, 2019. URL https://openreview.net/forum?id=B1gabhRcYX.

[31] Fengrui Tian, Shaoyi Du, and Yueqi Duan. MonoNeRF: Learning a generalizable dynamic radiance field from monocular videos. In Proceedings of the International Conference on Computer Vision (ICCV), October 2023.

[32] Jianyuan Wang, Christian Rupprecht, 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.

[33] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025.

[34] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. arXiv preprint arXiv:2507.13347, 2025.

[35] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

[36] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. SynSin: End-to-end view synthesis from a single image. In CVPR, 2020.

[37] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In CVPR, 2025.

[38] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2025.

[39] Botao Ye, Sifei Liu, Xueting Li, and Ming-Hsuan Yang. Self-supervised super-plane for neural 3d reconstruction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 21415–21424, 2023.

[40] Botao Ye, Sifei Liu, Haofei Xu, Li Xueting, Marc Pollefeys, Ming-Hsuan Yang, and Peng Songyou. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. arXiv preprint arXiv:2410.24207, 2024.

[41] Botao Ye, Boqi Chen, Haofei Xu, Daniel Barath, and Marc Pollefeys. Yonosplat: You only need one model for feedforward 3d gaussian splatting. In International Conference on Learning Representations (ICLR), 2026.

[42] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In ICCV, 2021.

[43] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In CVPR, 2021.

[44] 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, 2022.

[45] Jason Y Zhang, Amy Lin, Moneish Kumar, Tzu-Hsuan Yang, Deva Ramanan, and Shubham Tulsiani. Cameras as rays: Pose estimation via ray diffusion. In International Conference on Learning Representations (ICLR), 2024.

[46] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. European Conference on Computer Vision, 2024.

[47] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In CVPR, 2018.

[48] Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views, 2025. URL https://arxiv.org/abs/2502.12138.

[49] Qitao Zhao, Amy Lin, Jeff Tan, Jason Y. Zhang, Deva Ramanan, and Shubham Tulsiani. Diffusionsfm: Predicting structure and motion via ray origin and endpoint diffusion. In CVPR, 2025.

[50] Qitao Zhao, Hao Tan, Qianqian Wang, Sai Bi, Kai Zhang, Kalyan Sunkavalli, Shubham Tulsiani, and Hanwen Jiang. E-rayzer: Self-supervised 3d reconstruction as spatial visual pre-training. In CVPR, 2026.

[51] Tinghui Zhou, Matthew A. Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 6612–6619, 2017. URL https://api.semanticscholar.org/CorpusID:11977588.