Fin3R: Fine-tuning Feed-forward 3D Reconstruction Models via Monocular Knowledge Distillation

Weining Ren[1] **Hongjun Wang**[1] **Xiao Tan**[2] **Kai Han**[1]*
[1] Visual AI Lab, The University of Hong Kong
[2] Department of Computer Vision Technology (VIS), Baidu Inc.
weining@connect.hku.hk, hjwang@connect.hku.hk
tanxiao01@baidu.com, kaihanx@hku.hk

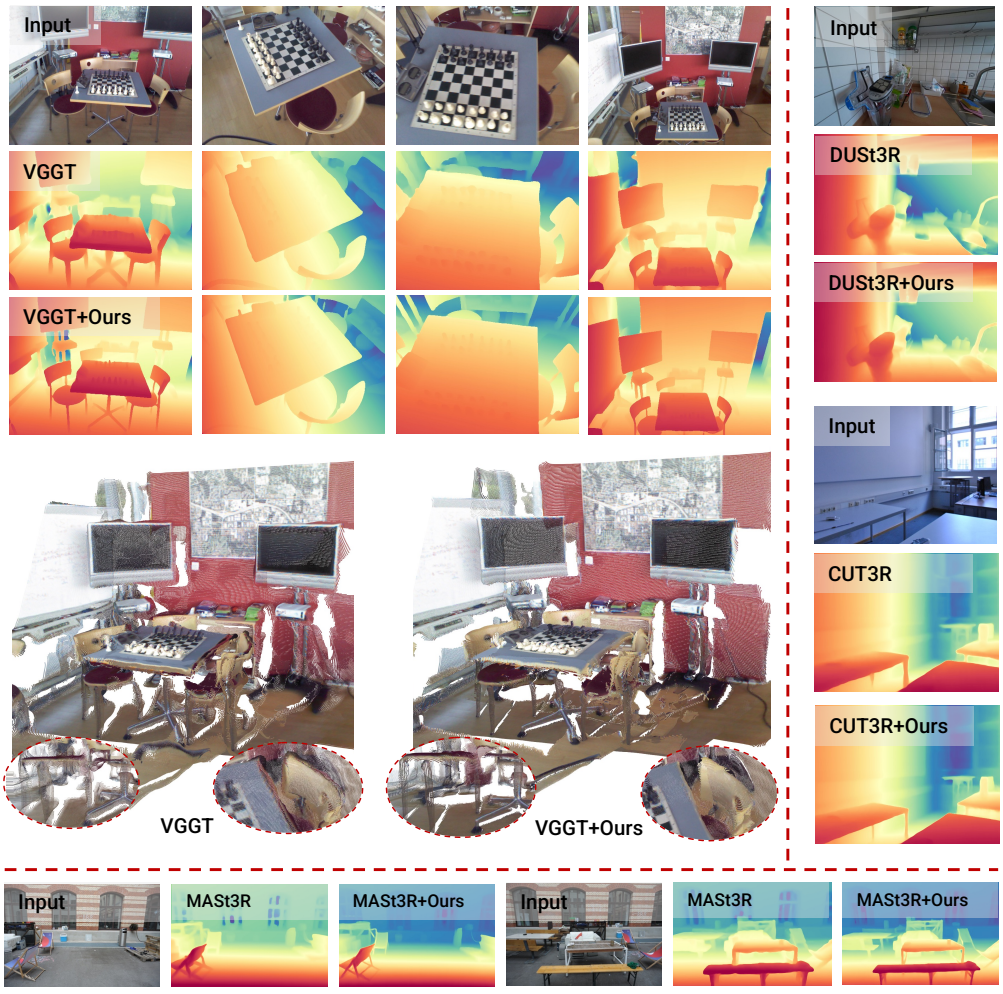


Figure 1: **Fin3R consistently improves the reconstructed geometry quality** in DUSt3R, MASt3R, CUT3R, and VGGT, recovering finer details and producing sharper boundaries.

*Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Abstract

We present Fin3R, a simple, effective, and general fine-tuning method for feed-forward 3D reconstruction models. The family of feed-forward reconstruction model regresses pointmap of all input images to a reference frame coordinate system, along with other auxiliary outputs, in a single forward pass. However, we find that current models struggle with fine geometry and robustness due to (*i*) the scarcity of high-fidelity depth and pose supervision and (*ii*) the inherent geometric misalignment from multi-view pointmap regression. Fin3R jointly tackles two issues with an extra lightweight fine-tuning step. We freeze the decoder, which handles view matching, and fine-tune only the image encoder—the component dedicated to feature extraction. The encoder is enriched with fine geometric details distilled from a strong monocular teacher model on large, unlabeled datasets, using a custom, lightweight LoRA adapter. We validate our method on a wide range of models, including DUSt3R, MASt3R, CUT3R, and VGGT. The fine-tuned models consistently deliver sharper boundaries, recover complex structures, and achieve higher geometric accuracy in both single- and multi-view settings, while adding only the tiny LoRA weights, which leave test-time memory and latency virtually unchanged. Project page: https://visual-ai.github.io/fin3r

1 Introduction

Recently, neural feed-forward 3D reconstruction models [68, 28, 64, 61, 75, 55, 87, 60] have demonstrated advantages in certain aspects compared to the traditional Structure from Motion (SfM) pipeline [48, 41]. These methods can transform a single image—or even hundreds of images—into pointmaps defined in the reference frame within a single forward pass, thereby eliminating the need for hand-crafted features and time-consuming iterative optimization. At their core, these architectures share a common structure: a shared encoder extracts features from input images, followed by a decoder correlating these features across views. Subsequent task-specific heads then regress pointmaps while optionally simultaneously estimating auxiliary outputs like camera parameters and depth.

Despite their efficiency and flexibility, these models still lag behind state-of-the-art monocular geometry estimation approaches [77, 66, 23, 43] in capturing fine geometric detail and robustness. While architectures such as CUT3R [65] leverage large-scale data supervision and VGGT [61] integrates gradient-based losses to capture fine details, the resulting depth and pointmap outputs remain coarse. Fine structures are frequently over-smoothed, object boundaries become blurred, and transparent or glossy surfaces are reconstructed with significant inaccuracies, yielding point clouds that lack crisp geometry. This persistent gap in performance raises a crucial question: why do these feed-forward models consistently struggle to capture high-fidelity geometry? To answer this, we identify two primary factors that limit the geometric fidelity of these models: (1) *Data quality constraints*: Current real-world datasets providing accurate camera poses and high-fidelity depth remain limited. Existing non-synthetic depth labels are noisy [77] and predominantly biased toward indoor environments. (2) *Long-sequence pointmap degradation*: Inherent ambiguities in multi-view pointmap regression impede the network's ability to capture fine details over long sequences.

Motivated by these challenges, we investigate whether extensive unlabelled single-view data can be used to fine-tune pre-trained models to improve fine geometry recovery and robustness without sacrificing multi-view performance. This approach relaxes the constraint of high-quality data and long-sequence degradation. Recalling the common structure of recent feed-forward reconstruction models, we distill a state-of-the-art monocular geometry estimator (MoGe [66]) into the encoder using the diverse SA-1B dataset [25], while freezing the decoder to preserve its multi-view performance.

However, we observe that naïve encoder-only distillation, though beneficial for single-frame accuracy, leads to an increase in encoder feature norms. This drift pushes the features outside the range expected by the frozen decoder and undermines multi-view capability. To counteract this, we initially combined LoRA [20] with multi-view data replay, but the shift persisted. We therefore embed customized re-normalization layers within each LoRA block to dynamically correct this drift. Our solution achieves crisp depth predictions for single images while maintaining multi-view performance, all without the need for additional decoder fine-tuning.

To summarize, we propose a simple, effective, and general fine-tuning approach. By freezing the decoder and integrating a customized re-normalization LoRA adapter into the encoder, we distill the model from a high-fidelity monocular teacher using a diverse dataset. Remarkably, the same implementation is applied to four baselines—DUSt3R's [68] pairwise prediction with relative depth, MASt3R's [28] pairwise prediction with metric depth, CUT3R's [64] recurrent network, and VGGT's [61] parallel transformer—yielding crisper and more robust single-view depth, while preserving or even slightly improving multi-view performance. Our contributions are threefold: (*i*) a general encoder-only distillation strategy that enhances local geometric detail and overall robustness in feed-forward 3D reconstruction models; (*ii*) a feature shift mitigation approach combining customized re-normalization LoRA with multi-view data replay to reduce distribution shifts over long sequences; and (*iii*) a comprehensive evaluation on DUSt3R, MASt3R, CUT3R, and VGGT, demonstrating improved depth fidelity and correspondence accuracy while preserving global multi-view performance.

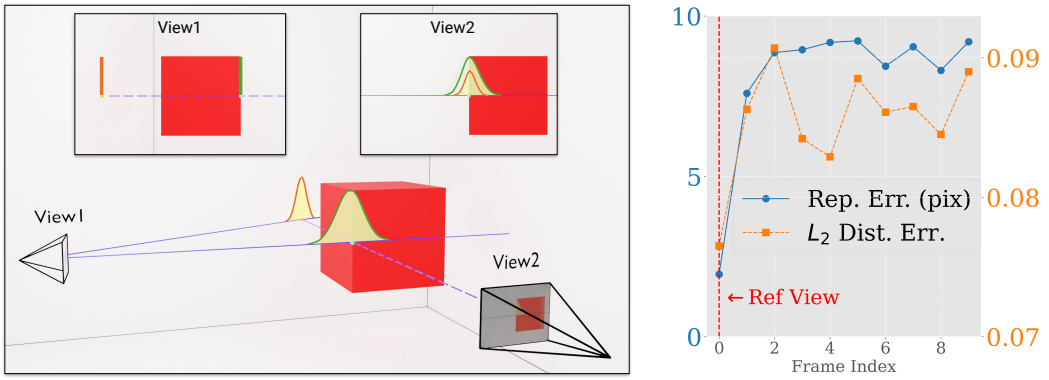
2 Related Work

Optimization-based Multi-view Reconstruction For over two decades, mainstream 3D reconstruction methods [19, 39] treated reconstruction as a large-scale optimization problem. The standard workflow [53, 1, 14] starts with exhaustive matching, triangulation, and bundle adjustment—*structure-from-motion* (SfM)—implemented in toolkits such as COLMAP [48]. SfM yields a sparse, metrically consistent point cloud that is densified by photo-consistent *multi-view stereo* (MVS). Early MVS relied on hand-crafted heuristics [16, 17]; recent variants adopt learned cost volumes [78, 18, 37] or neural-implicit global optimisation [38, 15]. Deep learning has also upgraded SfM components: keypoints [82, 12], matchers [47, 31], even the full loop via differentiable Bundle Adjustment (BA) [57, 62]. Yet these optimization-heavy pipelines remain calibration-sensitive and slow.

Feed-forward 3D Reconstruction Models Recent work removes explicit optimisation loops and predicts scene geometry in a single network pass. DUSt3R [68] pioneers this trend: from two uncalibrated images it produces a dense *PointMap* anchored in the first view, from which pose, depth, and correspondences are recovered through post-processing. MASt3R [28] retains the same backbone but adds feature heads for matching. To extend beyond pairs, methods diverge into (i) recurrent architectures that process frames sequentially, e.g., CUT3R [65] and Span3R [60], and (ii) fully parallel attention across all views, e.g., MV-DUSt3R++ [55], FLARE [87], Fast3R [76], and VGGT [61]. Despite their strong implicit multi-view correspondence capability, these feed-forward models still struggle to capture sharp local geometry and reconstruct complex surfaces. Recent fine-tuning works [35, 86] rely on test-time optimization, requiring per-scene finetuning for each new instance. In contrast, our method involves a single, universal finetuning phase to create one model that generalizes to new scenes in a zero-shot manner.

Monocular Priors for Multi-view Geometry Leveraging monocular cues to assist multi-view problems has a long history. Dense monocular depth, surface normals and semantics have been used to assist SLAM [56, 9, 33, 32, 89], to fill in gaps in dense reconstruction [44, 36, 72, 85], and to guide novel view synthesis [54, 74]. Other works explore monocular priors for relative pose [4, 84], PnP-RANSAC on depth maps [32], and monocular-assisted SfM [42]. More recently, significant progress in monocular depth [77, 43, 23, 66] and normal estimation [83, 73, 3] has positioned these methods as strong priors for a variety of tasks.

Recently, monocular priors have also been injected into the new trend of feed-forward 3D reconstruction networks. Align3R [34], Pow3R [21], and Mono3R [29] inject single-image depth (or sparse depth hints) to improve the pointmap prediction of DUSt3R-style models. However, they either (i) rely on an external geometric estimator or (ii) assume sparse, high-quality depth inputs. In contrast, our approach keeps the feed-forward pipeline *fully self-contained*: we do not introduce any extra heavy inference modules, or runtime overhead. Instead, we focus on training a stronger encoder that yields markedly more robust multi-view geometry without compromising speed.



(a) Scale Uncertainty Illustration

(b) Error metrics.

Figure 2: **Analysis of scale uncertainty and error metrics.** (a) Two views of a red cube are connected by a blue epipolar line. Gaussian distributions overlaid on a foreground point (green) and a background point (yellow) illustrate their respective scale uncertainties, with the foreground exhibiting notably larger epipolar dispersion after projection. (b) Reprojection error and Euclidean distance loss are computed for 10 inputs processed by VGGT [61], with 1,000 samples drawn from Hypersim.

3 Method

3.1 Observations and Challenges

Our analysis reveals two main challenges in existing datasets and long sequence scenarios that critically affect training of geometry regression heads:

Data Scarcity. Existing datasets suffer from depth [77] and pose noise, and the limited availability of multi-view data further restricts the model's ability to generalize. These noisy and insufficient labels hinder the model's capacity to capture fine details and to robustly adapt to diverse scenarios.

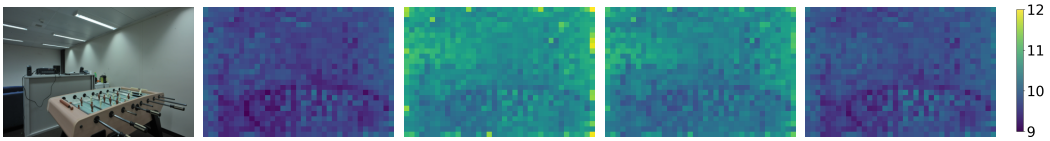
Long-Sequence Degradation. Long sequences introduce additional issues for pointmap regression: (1) *Coupled Prediction:* Although DUSt3R [68]'s multi-view pointmap regression has enabled feed-forward 3D reconstruction, it inherently couples pose and depth estimation in pointmap regression, injecting pose regression error into the geometry heads. (2) *Drift:* As the views progressively move further away from the initial reference frame, progressive drift becomes inevitable. This drift results in increasing errors on non-reference views and negatively affects the preservation of fine structural details. (3) *Scale Uncertainty:* During training, both predicted and ground-truth pointmap require normalization to ensure scale consistency.[2] However, this scale uncertainty tends to erode fine foreground boundary along the epipolar line in views beyond the first frame. This phenomenon is illustrated in Figure 2a and mathematically validated in the appendix.

Consequently, pointmap regression introduces substantial errors in non-reference views, as evidenced by the pronounced reprojection error in Figure 2b. Although CUT3R [65] leverages extensive depth supervision and VGGT [61] employs gradient-based loss to refine local geometry—with both methods incorporating dedicated self-view pointmap or depth estimation heads—the resulting outputs remain relatively coarse. We suspect that the multi-view pointmap regression undermines the performance of these self-view estimation heads, thereby limiting the model's ability to capture fine-grained details.

These observations not only underscore the necessity for high-quality supervision from diverse datasets but also highlight the inherent challenges associated with multi-view pointmap regression.

3.2 Fin3R

[2]Although VGGT [61] circumvents an explicit normalization step by implicitly inferring the prediction scale, it does not entirely resolve the inherent scale uncertainty in SfM.



(a) Input Image (b) VGGT Avg: 9.61 (c) LoRA Only Avg: 10.53 (d) LoRA+Replay Avg: 10.34 (e) Full Avg: 9.73

Figure 3: **Heatmaps show spatial variations in L_2 norms of encoder patch tokens across configurations.** "Avg" is the average norm of the feature map, and (e) Full indicates the full model with re-normalization LoRA and multi-view data replay.

Based on these observations, we introduce Fin3R—our solution that integrates a lightweight fine-tuning stage to simultaneously address both challenges by monocular knowledge distillation.

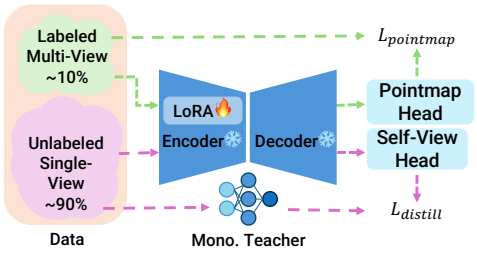


Figure 4: **Pipeline of our method.** Green dashed lines denote pointmap supervision; purple dashed lines indicate distillation supervision.

3.2.1 Encoder-only Distillation

We aim to enhance our model's capability to capture fine details and complex surface geometries while preserving its multi-view performance. Recall that feed-forward 3D reconstruction models typically consist of a shared encoder, which extracts features from input images, followed by a decoder that correlates these features across views. We contend that the limitations in detail recovery primarily originate from the encoder. Therefore, we enrich the encoder using a robust monocular teacher [66], distilled on a large and diverse dataset [25]. This strategy is designed to improve local geometric detail recovery without compromising the decoder's proven matching capabilities.

3.2.2 Monocular Finetuning Needs Feature Re-normalization

An initial exploration into naïve encoder distillation—through full parameter fine-tuning—showed that while the single-view geometric details were significantly improved, the fine-tuning adversely affected the multi-view matching capability even when we froze the decoder. Our early attempts to alleviate this issue involved leveraging LoRA and multi-view data replay. However, these strategies only partially mitigated the problem, as the degradation in multi-view performance persisted.

A closer examination of the model revealed a key culprit: single-view distillation led to a continuous increase in feature norms, as shown in Figure 3. This norm shift pushed the feature beyond the range expected by the frozen decoder, thereby impairing multi-view matching. To directly address this challenge, we propose a refined integration of LoRA with a re-normalization strategy specifically designed to constrain feature norm drift. Concretely, given an original weight matrix W and its corresponding LoRA update ΔW, we re-normalize the combined weight after each update as follows:

$$W' = \frac{(W + \Delta W) \cdot \|W\|_2}{\|W + \Delta W\|_2}.$$

Here, $\|\cdot\|_2$ denotes the L2 norm. This operation ensures that the updated weight W' maintains the original norm $\|W\|_2$, thereby preserving the distribution of feature activations that the frozen decoder expects. As a result, we retain the crucial multi-view matching capability while still obtaining the benefits of enhanced local geometry recovery from self-view distillation. Although this method is not necessarily sufficient to address all feature shifts, we found it generally effective in most cases.

3.3 Training

We optimize two loss functions computed over images indexed by i in the training set. For each image, the monocular distillation loss refines single-view details by aligning the predicted depth D_i with the high-fidelity pseudo-label $\hat{D}_i$ provided by a monocular teacher, weighted by the aleatoric uncertainty β_i^D; it is defined as $\mathcal{L}_{\text{distill}}^{(i)} = \beta_i^D \|D_i - \hat{D}_i\|_2^2 - \lambda \log \beta_i^D$. The pointmap regression loss

enforces robust multi-view matching while mitigating potential feature shift; to ensure this loss is applied only to multi-view samples, we introduce an indicator function $\mathbf{1}_{\text{mv}}(i)$ that equals 1 if the i-th image belongs to the multi-view dataset and 0 otherwise, and define the loss as $\mathcal{L}^{(i)}_{\text{pointmap}} = \mathbf{1}_{\text{mv}}(i)\left(\beta_i^P \|P_i - P_i^{\text{GT}}\|_2^2 - \lambda \log \beta_i^P\right)$. The overall training objective is the average loss over all N images, given by $\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathcal{L}^{(i)}_{\text{distill}} + \mathcal{L}^{(i)}_{\text{pointmap}}\right)$, with the uncertainty terms modeled as in [24].

4 Experiment

Implementation Details. We use MoGe [66] as the teacher model for pseudo-label generation. Since the depth predicted by MoGe is affine-invariant, we subtract the shift in the z-component and then apply the normalization used in DUSt3R. For DUSt3R [68], we use 2-view data with distillation supervision applied exclusively to the view-1 pointmap head for distillation loss. In contrast, CUT3R [65] and VGGT [61] utilize 2–8 views, with supervision on either the self-view head or the depth head. During each epoch, we sample 20,000 images from SA-1B [26], 1,000 from Hypersim [46], and 1,000 from TartainAir [69]. Training runs for 10 epochs on four NVIDIA L20 GPUs over a single day. Further implementation details are provided in the appendix.

Evaluation Protocol. We evaluate our approach across three settings: single-view, two-view, and multi-view. In the single-view setting, we focus on monocular depth estimation. The two-view configuration evaluates relative pose estimation, where we extract pairwise correspondences using DUSt3R [68]'s matching method for VGGT. In the multi-view setting, we perform multi-view depth estimation, pointmap estimation, and pose estimation. Since CUT3R [65] is designed for long sequences and unsuitable for pairwise correspondences, we remove it in the two-view evaluation.

Table 1: **Quantitative results for monocular depth estimation.** "+Ours" denotes the integration of our fine-tuning, and MoGe is the teacher model. Best results in each session are highlighted in **bold**.

Method	NYUv2		KITTI		ETH3D		iBims-1		DDAD		DIODE		HAMMER		*Average*	
	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$	Rel ↓	$\delta_1^\uparrow$
Scale-invariant relative depth																
DUSt3R [68]	3.83	97.7	7.64	91.1	5.35	95.9	3.97	96.5	17.34	75.5	6.85	92.4	4.23	96.9	7.03	92.3
DUSt3R+Ours	**3.68**	**97.8**	**6.02**	**94.7**	**4.41**	**96.8**	**3.47**	**97.4**	**13.11**	**83.1**	**4.70**	**95.3**	**3.66**	**98.7**	**5.58**	**94.8**
CUT3R [65]	3.73	97.9	7.20	91.7	4.69	96.4	4.06	96.4	15.62	76.9	5.93	93.2	4.01	98.2	6.46	92.9
CUT3R+Ours	**3.68**	**97.9**	**5.93**	**94.7**	**4.67**	**96.6**	**3.46**	**97.7**	**13.12**	**82.3**	**5.08**	**94.8**	**3.20**	**99.3**	**5.59**	**94.7**
VGGT [61]	3.14	98.3	5.83	94.1	3.64	97.5	3.61	96.8	13.74	81.3	5.24	94.5	5.18	95.2	5.77	94.0
VGGT+Ours	**3.10**	**98.3**	**4.59**	**97.2**	**3.07**	**98.7**	**2.73**	**98.2**	**10.65**	**88.1**	**3.59**	**96.7**	**2.31**	**99.5**	**4.29**	**96.7**
MoGe [66]	3.02	98.5	4.39	97.4	2.96	98.9	2.65	98.2	9.64	90.0	3.23	97.4	3.09	98.2	4.14	96.9
Metric depth																
MASt3R [28]	**10.79**	**89.6**	55.11	10.9	46.91	21.3	18.65	61.5	62.90	4.3	55.34	18.3	97.62	5.6	49.62	30.2
MASt3R+Ours	11.71	88.4	**10.69**	**89.1**	**26.30**	**56.0**	**11.29**	**86.3**	**26.50**	**55.5**	**22.84**	**50.1**	**83.89**	**24.5**	**27.60**	**64.3**
MoGe-2 [67]	6.92	96.7	16.72	70.1	10.92	88.2	14.08	81.1	15.82	74.1	15.97	71.3	23.30	68.5	14.82	78.6

4.1 Monocular Depth Estimation

We follow the evaluation of MoGe [66] to evaluate our method using standard metrics: relative absolute difference (rel) and the δ_1 score. Specifically, $\text{rel} = \frac{1}{N}\sum_{i=1}^{N}\frac{|d_i - d_i^*|}{d_i^*}$, where d_i and d_i^* denote the predicted and ground truth depths, respectively, while δ_1 represents the percentage of predictions satisfying $\max\left(\frac{d_i}{d_i^*}, \frac{d_i^*}{d_i}\right) < 1.25$. Table 1 presents quantitative results for affine-invariant depth evaluation. The table shows that our integrated models consistently achieve lower relative depth error and higher δ_1 scores. Figure 5 shows the qualitative comparison between baselines and the results from our fine-tuning method. After fine-tuning, our method improves the model's ability to capture fine details and complex surfaces such as transparent ones. Fine-tuned VGGT performs almost as well as the state-of-the-art expert model, MoGe. Interestingly, we observe that although DUSt3R's depth estimates rank last among the evaluated models, they exhibit the sharpest boundaries compared with the other two baseline models. This is likely because CUT3R and VGGT are trained on long sequences and are consequently more affected by the long-sequence degradation

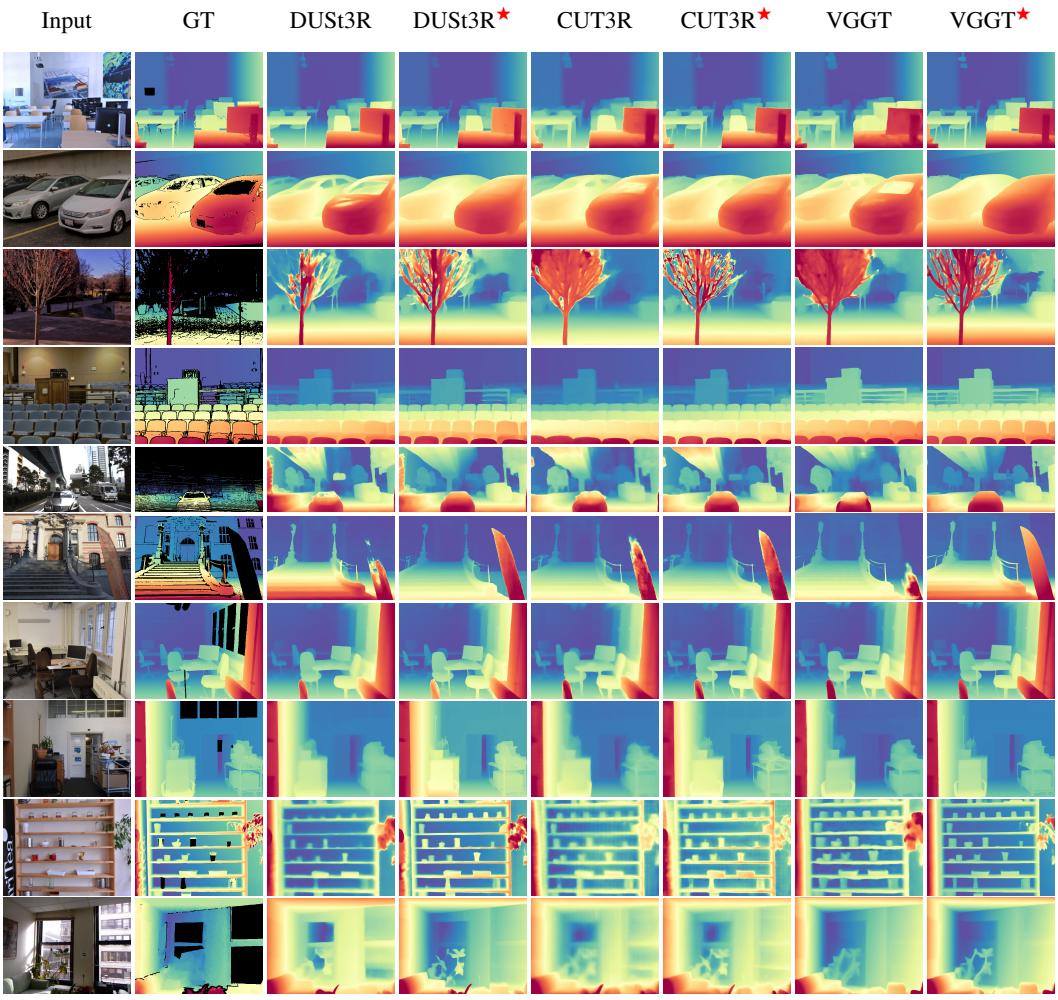


Figure 5: **Depth prediction across baseline models.** ★ indicates integration with our method.

discussed in Section 3.1. We also present the fine-tuned MASt3R model with metric depth prediction, demonstrating that our method is capable of handling not only relative depth prediction but also metric depth estimation.

4.2 Relative Pose Estimation

Table 2 summarizes our evaluation of relative pose estimation on the ScanNet dataset [10]. Following [47], we assess performance using area-under-the-curve (AUC) metrics computed at thresholds of 5, 10, and 20 degrees. The results indicate that our fine-tuning method consistently improves the baseline model correspondence by improving the geometry. In particular, our fine-tuned VGGT model outperforms Reloc3r [11] at the 5° threshold, despite Reloc3R being designed only for pose regression and lacking geometric modeling capability.

4.3 Multi-view Depth Estimation

Following CUT3R [65], we evaluate the performance of our method on video depth estimation. Table 4 summarizes the results, demonstrating that our method preserves multi-view consistency and improves single-view accuracy. The fine-tuned versions of CUT3R and VGGT consistently outperform their respective baselines across datasets spanning diverse domains. Note that VGGT is not trained on dynamic datasets, so its performance bottleneck may stem from dataset limitations rather than our fine-tuning method.

Table 2: **Relative Camera Pose Evaluation on the ScanNet1500 [10, 47] datasets.** "Ours" indicates the integration of our distillation method. Better results are highlighted in **bold**.

Methods	ScanNet1500		
	AUC@5	AUC@10	AUC@20
Efficient LoFTR [70]	19.20	37.00	53.60
ROMA [13]	28.90	50.40	68.30
NoPoSplat [80]	31.80	53.80	71.70
DUSt3R [68]	31.61	53.77	70.99
DUSt3R+Ours	**33.73**	**55.67**	**72.66**
MASt3R [28]	37.60	59.96	76.24
MASt3R+Ours	**37.93**	**60.21**	**76.68**
VGGT [61]	28.40	47.36	61.51
VGGT+Ours	**35.21**	**56.70**	**72.80**
Reloc3r [11]	34.79	58.37	75.56

Table 3: **Quantitative Results for Multi-view Pose Estimation on RealEstate10k [88].** "Ours" is the fine-tuned model using our method. Better results are highlighted in **bold**.

Methods	RealEstate10k		
	RRA@5	RTA@5	AUC@30
DUSt3R [68]	94.01	42.39	62.40
DUSt3R+Ours	**95.41**	**47.07**	**64.81**
MASt3R [28]	94.89	52.21	73.45
MASt3R+Ours	**95.02**	**53.74**	**73.87**
CUT3R [65]	96.66	61.66	78.95
CUT3R+Ours	**96.99**	**62.15**	**79.13**
VGGT [61]	95.28	53.14	74.18
VGGT+Ours	**96.27**	**56.54**	**75.35**

Table 4: **Results for Video Depth Estimation**. The arrows (↓/↑) indicate whether lower or higher values are better. Best results are highlighted in **bold**.

Method	**ETH3D** [49]		**T&T** [27]		**KITTI** [58]		**Sintel** [6]		**Bonn** [40]	
	rel ↓	δ_1 ↑	rel ↓	δ_1 ↑	rel ↓	δ_1 ↑	rel ↓	δ_1 ↑	rel ↓	δ_1 ↑
CUT3R [65]	**0.126**	**83.1**	0.209	69.5	0.123	87.4	0.428	47.4	0.077	93.9
CUT3R+Ours	0.130	82.8	**0.180**	**76.2**	**0.112**	**89.8**	**0.406**	**58.4**	**0.062**	**96.8**
VGGT [61]	0.044	97.9	0.137	85.3	0.072	96.5	0.301	68.4	0.052	97.3
VGGT+Ours	**0.041**	**99.2**	**0.115**	**88.0**	**0.069**	**96.6**	**0.252**	**72.7**	**0.048**	**97.5**

4.4 Multi-view Pointmap Estimation

Table 5 presents the multi-view reconstruction performance on the 7Scenes [52] and NRGBD [2] datasets following Spann3R [60]. Note that DUSt3R employs global alignment, while the other methods operate in a feed-forward manner. Because both DUSt3R and VGGT produce scale-invariant point maps, we apply Umeyama alignment [59] to align scale. We report mean and median values for three metrics: accuracy (Acc), completeness (Comp), and normal consistency (NC). The results indicate that models enhanced with our distillation method consistently achieve lower Acc and Comp as well as improved NC scores across most baselines. Qualitative results can be found at Figure 6.

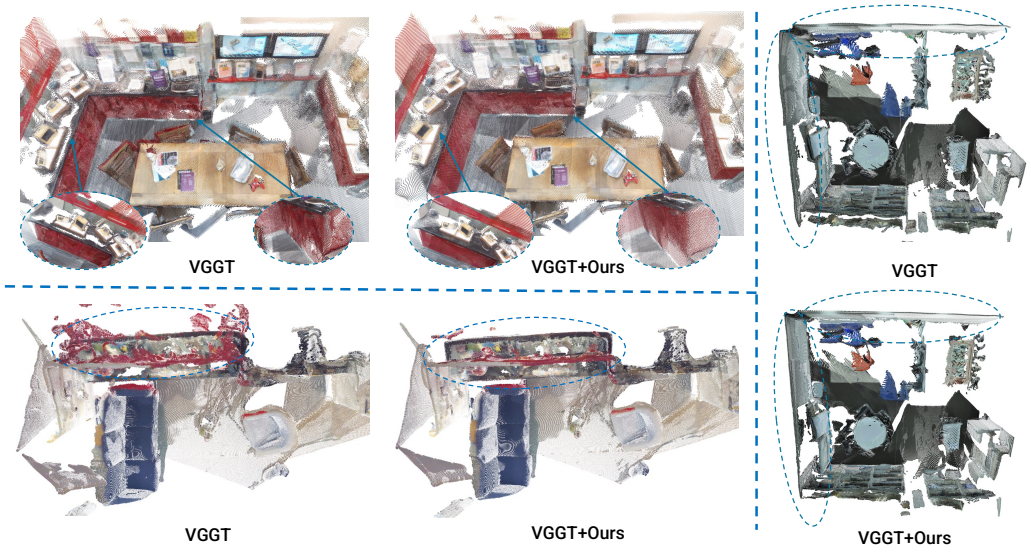


Figure 6: **Additional results of pointmap estimation**.

Table 5: **Pointmap Regression on on 7-Scenes [52] and NRGBD [2] Datasets.** "+Ours" represents the integration of our distillation method. The best results at each session are in **bold.**

	7-Scenes [52]						NRGBD [2]					
	Acc↓		Comp↓		NC↑		Acc↓		Comp↓		NC↑	
Method	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.
DUSt3R [68]	0.026	0.011	0.033	0.018	0.641	0.725	0.050	0.030	0.036	0.019	0.851	0.983
DUSt3R+Ours	**0.024**	**0.009**	**0.029**	**0.015**	**0.641**	**0.726**	**0.043**	**0.027**	**0.030**	**0.017**	**0.863**	**0.986**
CUT3R [65]	**0.024**	**0.011**	0.029	0.010	0.664	0.758	0.075	0.031	0.046	0.019	0.828	0.966
CUT3R+Ours	0.025	0.012	**0.026**	**0.010**	**0.666**	**0.762**	**0.075**	**0.028**	**0.043**	**0.019**	**0.833**	**0.968**
VGGT [61]	0.017	0.006	0.024	0.011	0.645	0.727	**0.019**	**0.012**	**0.018**	**0.009**	0.914	0.992
VGGT+Ours	**0.012**	**0.006**	**0.023**	**0.011**	**0.651**	**0.739**	0.021	0.014	0.020	0.011	**0.921**	**0.993**

Table 6: **Pointmap Regression on the DTU and ETH3D datasets**. The arrows (↓/↑) indicate whether lower or higher values are better. Best results are highlighted in **bold**.

Method	DTU [22]						ETH3D [50]					
	Acc. ↓		Comp. ↓		N.C. ↑		Acc. ↓		Comp. ↓		N.C. ↑	
	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.
Pi3 [71]	1.151	0.622	**1.793**	**0.629**	0.668	0.754	**0.194**	0.130	0.220	0.135	**0.867**	0.965
VGGT [61]	1.187	0.715	2.229	1.309	0.694	0.779	0.290	0.196	0.371	0.230	0.839	0.932
VGGT+Ours	**0.948**	**0.520**	1.879	0.905	**0.699**	**0.787**	0.209	**0.112**	**0.170**	**0.085**	0.861	**0.972**

Table 7: **Ablation Study for Distillation Module.** Combining all our strategies yields the highest accuracy.

Label Supv.	Mono. Teacher	SA-1B	Rel (↓)	δ_1 (↑)	Acc (↓)
✗	✗	✗	5.68	94.1	0.017
✓	✗	✗	5.21	95.0	0.014
✗	✓	✗	5.00	95.3	0.013
✗	✓	✓	**4.35**	**96.3**	**0.012**

Table 8: **Ablation Study on Fine-tuning Strategy.** Our proposed components consistently improve matching performance.

Method	AUC@5	AUC@10	AUC@20
VGGT	28.40	47.36	61.51
(1) +Dec. Full	28.42	51.59	67.30
(2) +Enc. Full	32.06	52.29	68.04
(3) +Enc.&Dec. Full	26.35	45.90	60.02
(4) +Enc. Lora	32.96	54.21	70.40
(5) +Enc. Lora+Re-norm	**35.21**	**56.70**	**72.80**

Table 6 also compares our method with the concurrent model Pi3 [71] on the DTU [22] and ETH3D [50] datasets using the pointmap head of VGGT. Our method delivers comparable performance while requiring significantly fewer resources for fine-tuning.

4.5 Multi-view Pose Estimation

Table 3 summarizes the performance of baseline models and our fine-tuned methods on the RealEstate10k [88] dataset. We evaluate performance using three metrics: Recall of Relative Angle (RRA@5), Recall of Relative Translation (RTA@5), and AUC@30. These results indicate that our method primarily refines the geometry head without significantly affecting the pose head. We attribute this improvement to the decoder functioning as an implicit feature matcher, which allows it to leverage the enhanced feature details for more accurate pose prediction.

4.6 Ablation Study

Distillation Strategy. Table 7 shows that our distillation pipeline incrementally enhances geometric accuracy on VGGT [61]. The first two columns report mean monocular depth metrics (see Table 1), while the final column details the 7-Scenes [52] accuracy. The top row represents VGGT model without fine-tuning, which can benefit from single-view distillation (second row) on a subset of training datasets (see appendix) with supervision from dataset depth labels. Replacing the depth labels with a monocular teacher further improves performance, and changing the dataset to SA-1B yields the best performance. Together, these results highlight that monocular finetuning with high-quality pseudo-labels from the diverse dataset improves both single-view and multi-view accuracy.

Finetuning Strategy. Table 8 evaluates our fine-tuning strategy on ScanNet relative pose estimation using VGGT. Lines (1), (2), and (3) demonstrate that fine-tuning the decoder with monocular data harms multi-view consistency, highlighting the effectiveness of our encoder-only fine-tuning design. Lines (2), (4), and (5) show that full-parametric fine-tuning improves the baseline's performance, while integrating the LoRA module further refines the representations. Notably, the re-normalization LoRA mitigates norm drift, leading to progressively improved matching performance. These results confirm that our modifications effectively reduce domain shifts while enhancing both fine detail recovery and multi-view consistency.

4.7 Discussion

Confidence and Fine Details: During our experiments, we observed that models like VGGT often produce blurry geometry accompanied by low confidence scores, as shown in Fig 8. After our fine-tuning, the model becomes more confident in its predictions and is capable of generating sharper geometry with better calibrated confidence. We attribute this improvement primarily to the incorporation of unlabeled datasets, which enhance the model's robustness and overall performance. This underscores the necessity of including in-the-wild data alongside high-quality datasets during training to achieve optimal results.

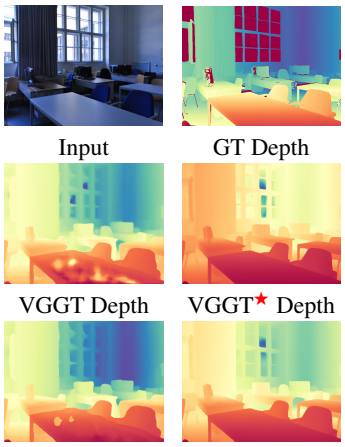


Figure 7: **Depth from depth head and pointmap head of VGGT.** ★ denotes our fine-tuning model.

Cross-Head Generalization via a Robust Encoder While our training only distills the depth head of VGGT with pseudo-label, our findings indicate that the pointmap head exhibits similar improvements (see Figure 7). This demonstrates that a robustly trained encoder benefits downstream heads even without direct supervision.

Position of Our Method Our approach is a lightweight, resource-efficient fine-tuning strategy for feed-forward reconstruction models. By carefully fine-tuning the encoder, it avoids the resource-intensive decoder tuning, which typically requires long-sequence inputs from diverse datasets with large batch sizes. Although further decoder tuning may yield additional gains, our method minimizes complexity without compromising quality. As 3D vision enters the era of large models, we hope our approach and analysis offer valuable insights into fine-tuning 3D large models with limited resources.

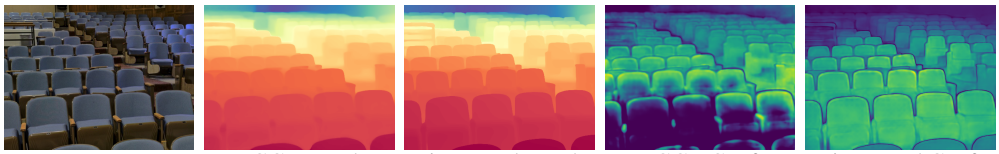


Figure 8: **Visualization of depth and confidence predictions**. The confidence color ranges from dark purple (low confidence) to bright yellow (high confidence).

5 Conclusion

We introduced Fin3R, a lightweight fine-tuning approach that leverages monocular distillation and re-normalization LoRA to enhance fine geometry and robustness in feed-forward 3D reconstruction models. Extensive experiments on DUSt3R, MASt3R, CUT3R, and VGGT validate that our method sharpens local details while preserving robust cross-view ability. Our results highlight the effectiveness and efficiency of our fine-tuning strategy, achieving notable performance gains while requiring minimal computational overhead.

Acknowledgments This work is supported by Hong Kong Research Grant Council - General Research Fund (Grant No. 17213825). Weining Ren is supported by Hong Kong PhD Fellowship Scheme (HKPFS).

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 2011. 3

[2] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, 2022. 8, 9

[3] Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *CVPR*, 2024. 3

[4] Daniel Barath and Chris Sweeney. Relative pose solvers using monocular depth. In *ICPR*, 2022. 3

[5] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *NeurIPS*, 2021. 23

[6] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 8

[7] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020. 23

[8] Yue Chen, Xingyu Chen, Anpei Chen, Gerard Pons-Moll, and Yuliang Xiu. Feat2gs: Probing visual foundation models with gaussian splatting. In *CVPR*, 2025. 24

[9] Jan Czarnowski, Tristan Laidlow, Ronald Clark, and Andrew J Davison. Deepfactors: Real-time probabilistic dense monocular slam. *RAL*, 2020. 3

[10] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 7, 8, 23

[11] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. *arXiv preprint arXiv:2412.08376*, 2024. 7, 8

[12] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *CVPR*, 2019. 3

[13] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. RoMa: Robust Dense Feature Matching. *CVPR*, 2024. 8

[14] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *ECCV*, 2010. 3

[15] Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *NeurIPS*, 2022. 3

[16] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015. 3

[17] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *ICCV*, 2015. 3

[18] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *CVPR*, 2020. 3

[19] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 3

[20] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 2

[21] Wonbong Jang, Philippe Weinzaepfel, Vincent Leroy, Lourdes Agapito, and Jerome Revaud. Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. *arXiv preprint arXiv:2503.17316*, 2025. 3

[22] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, 2014. 9

[23] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. 2, 3

[24] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *NeurIPS*, 2017. 6

[25] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 2, 5, 23

[26] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 6, 23

[27] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM TOG*, 2017. 8

[28] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024. 2, 3, 6, 8

[29] Wenyu Li, Sidun Liu, Peng Qiao, and Yong Dou. Mono3r: Exploiting monocular cues for geometric 3d reconstruction. *arXiv preprint arXiv:2504.13419*, 2025. 3

[30] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 23

[31] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. *arXiv preprint arXiv:2306.13643*, 2023. 3

[32] Sheng Liu, Xiaohan Nie, and Raffay Hamid. Depth-guided sparse structure-from-motion for movies and tv shows. In *CVPR*, 2022. 3

[33] Shing Yan Loo, Syamsiah Mashohor, Sai Hong Tang, and Hong Zhang. Deeprelativefusion: Dense monocular slam using single-image relative depth prediction. In *IROS*, 2021. 3

[34] Jiahao Lu, Tianyu Huang, Peng Li, Zhiyang Dou, Cheng Lin, Zhiming Cui, Zhen Dong, Sai-Kit Yeung, Wenping Wang, and Yuan Liu. Align3r: Aligned monocular depth estimation for dynamic videos. *arXiv preprint arXiv:2412.03079*, 2024. 3

[35] Ziqi Lu, Heng Yang, Danfei Xu, Boyi Li, Boris Ivanovic, Marco Pavone, and Yue Wang. Lora3d: Low-rank self-calibration of 3d geometric foundation models. *arXiv preprint arXiv:2412.07746*, 2024. 3

[36] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *ACM TOG*, 2020. 3

[37] Zeyu Ma, Zachary Teed, and Jia Deng. Multiview stereo with cascaded epipolar raft. In *ECCV*, 2022. 3

[38] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*, 2020. 3

[39] Onur Özyeşil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion*. *Acta Numerica*, 26:305–364, 2017. 3

[40] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *IROS*, 2019. 8

[41] Linfei Pan, Daniel Barath, Marc Pollefeys, and Johannes Lutz Schönberger. Global Structure-from-Motion Revisited. In *ECCV*, 2024. 2

[42] Zador Pataki, Paul-Edouard Sarlin, Johannes L. Schönberger, and Marc Pollefeys. MP-SfM: Monocular Surface Priors for Robust Structure-from-Motion. In *CVPR*, 2025. 3

[43] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *CVPR*, 2024. 2, 3

[44] Matia Pizzoli, Christian Forster, and Davide Scaramuzza. Remode: Probabilistic, monocular dense reconstruction in real time. In *ICRA*, 2014. 3

[45] 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 *ICCV*, 2021. 23, 24

[46] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, 2021. 6, 23

[47] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, pages 4938–4947, 2020. 3, 7, 8

[48] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 2, 3

[49] Thomas Schöps, Torsten Sattler, and Marc Pollefeys. BAD SLAM: Bundle adjusted direct RGB-D SLAM. In *CVPR*, 2019. 8

[50] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, pages 3260–3269, 2017. 9

[51] Philipp Schröppel, Jan Bechtold, Artemij Amiranashvili, and Thomas Brox. A benchmark and a baseline for robust multi-view depth estimation. In *3DV*, 2022. 23

[52] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pages 2930–2937, 2013. 8, 9

[53] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 3

[54] Jiuhn Song, Seonghoon Park, Honggyu An, Seokju Cho, Min-Seop Kwak, Sungjin Cho, and Seungryong Kim. Därf: boosting radiance fields from sparse inputs with monocular depth adaptation. In *NeurIPS*, 2023. 3

[55] Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. *arXiv preprint arXiv:2412.06974*, 2024. 2, 3

[56] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *CVPR*, 2017. 3

[57] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *NeurIPS*, 2021. 3

[58] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *3DV*, 2017. 8

[59] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE TPAMI*, 1991. 8

[60] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024. 2, 3, 8

[61] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, 2025. 2, 3, 4, 6, 8, 9, 24

[62] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *CVPR*, 2024. 3, 23

[63] Jianyuan Wang, Christian Rupprecht, and David Novotny. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *ICCV*, pages 9773–9783, 2023. 24

[64] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. *arXiv preprint arXiv:2501.12387*, 2025. 2, 3

[65] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state, 2025. 2, 3, 4, 6, 7, 8, 9, 24

[66] Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. *arXiv preprint arXiv:2410.19115*, 2024. 2, 3, 5, 6, 23, 29

[67] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *NeurIPS*, 2025. 6

[68] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 2, 3, 4, 6, 8, 9, 24

[69] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. *IROS*, 2020. 6, 23

[70] Yifan Wang, Xingyi He, Sida Peng, Dongli Tan, and Xiaowei Zhou. Efficient LoFTR: Semi-dense local feature matching with sparse-like speed. In *CVPR*, 2024. 8

[71] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Scalable permutation-equivariant visual geometry learning. *arXiv preprint 2507.13347*, 2025. 9

[72] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *ICCV*, 2021. 3

[73] Guangkai Xu, Yongtao Ge, Mingyu Liu, Chengxiang Fan, Kangyang Xie, Zhiyue Zhao, Hao Chen, and Chunhua Shen. What matters when repurposing diffusion models for general dense perception tasks? *arXiv preprint arXiv:2403.06090*, 2024. 3

[74] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *CVPR*, 2025. 3

[75] 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 *CVPR*, June 2025. 2

[76] 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. *arXiv preprint arXiv:2501.13928*, 2025. 3

[77] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 2, 3, 4

[78] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, 2018. 3

[79] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *CVPR*, 2020. 23

[80] 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. 8

[81] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023. 23

[82] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. LIFT: Learned Invariant Feature Transform. In *ECCV*, 2016. 3

[83] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *ICCV*, 2023. 3

[84] Yifan Yu, Shaohui Liu, Rémi Pautrat, Marc Pollefeys, and Viktor Larsson. Relative pose estimation through affine corrections of monocular depth priors. In *CVPR*, 2025. 3

[85] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. In *NeurIPS*, 2022. 3

[86] Yuheng Yuan, Qiuhong Shen, Shizun Wang, Xingyi Yang, and Xinchao Wang. Test3r: Learning to reconstruct 3d at test time. *NeurIPS*, 2025. 3

[87] 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. *CVPR*, 2025. 2, 3

[88] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM TOG*, 2018. 8, 9, 23

[89] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. In *3DV*, 2024. 3

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction clearly reflect the main contribution of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Our custom re-normalization Lora can not be merged into models like vanilla Lora. Although our method enhances fine geometric details, its boundary accuracy remains inferior to that of MoGe, which produces sharper results. A mixed-teacher strategy or a more dedicated distillation design may offer further improvements. Additionally, the current model supports only 512/518 resolution, and scaling to higher resolutions remains a challenge for future work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Our method is simple and easy to integrate to any baselines. We provide all details need to reproduce our results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We will open-source the code and the model after the paper is accepted.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All details for training and testing are included in the supplemental material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The error bar is not common in feed-forward 3D reconstruction.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: It's stated in the Experiment Section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We follow the ethics of NeurIPS in every respect.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This research enhances 3D reconstruction, benefiting robotics, AR/VR, and accessibility by using diverse single-view data. However, improved realism carries risks of misuse in generating synthetic content, and potential biases from teacher models or datasets must be addressed. Responsible development and ethical guidelines are crucial to navigate these impacts. This work contributes to more detailed and robust 3D perception.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We follow the license of VGGT, using Attribution-NonCommercial 4.0 International license.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA] .

    Justification: Paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA] ,

    Justification: LLM is only used for writing and editing.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

Appendix

A Experiment Details

A.1 Training Details

In all experiments, we set both the rank and alpha of LoRA to 8.

DUSt3R. Since DUSt3R doesn't have a dedicated self-view head for canonical view estimation, we use DUSt3R's first viewpoint pointmap regression head for distillation. Training is performed at a resolution of 512 width, with aspect ratios (e.g., 16:9, 4:3) randomly sampled for each batch. During each epoch, we randomly sample 20,000 pairs from the SA-1B [25] dataset, 1,000 pairs from the Hypersim [46] dataset, and 1,000 pairs from the TartanAir [69] dataset. The model is fine-tuned for 10 epochs. The learning rate is initialized at 1e-4 with a one-epoch warm-up phase and is gradually decayed to a minimum of 1e-6. A batch size of 2 per GPU is used, and gradients are accumulated over 8 iterations to achieve an effective batch size of 64.

CUT3R/VGGT. We compute the distillation loss using the self-view pointmap head for CUT3R and the depth head for VGGT, following the same dataset configuration as in DUSt3R fine-tuning. CUT3R is trained at a resolution of 512 width, while VGGT is trained at a resolution of 518 width. The model is fine-tuned for 10 epochs with an initial learning rate of 1e-4, which is warmed up for one epoch and then gradually decayed to a minimum of 1e-6. Additionally, the sequence length is dynamically selected between 2 and 8, with the product of batch size and sequence length fixed at 8. The accumulation iteration is changed accordingly to ensure an effective total batch size of 64.

A.2 Evaluation Details

Monocular Depth Estimation. We follow the evaluation protocol from MoGe [66] to assess our models. For DUSt3R, we duplicate the input images and use the z value from the view-1 pointmap head as the predicted depth. For CUT3R, depth is obtained from the z value of the self-view pointmap head, and for VGGT, we use the output of the depth head. Since these models are trained at resolutions of 512 width (or 518 width for VGGT), the original images are resized accordingly for evaluation. Although this differs from the standard MoGe protocol, which evaluates at higher resolutions, we ensure that both the base model and our fine-tuned models share the same settings. Furthermore, we exclude evaluation datasets such as Sintel and Spring since DUSt3R and VGGT are not designed for dynamic scenes.

Two-view Evaluation. We extract two-view correspondences using the nearest neighbor matching strategy from DUSt3R, which leverages geometric distance and is well-suited for assessing our enhanced geometry. We avoid using VGGT's tracking head for matching for two main reasons. First, the current release of VGGT's tracking head does not perform as well as the version reported in the original paper[3]. Second, in the Scannet-1500 relative pose estimation task, our geometry-based correspondence method outperforms the tracking-based approach described in the original VGGT paper. Furthermore, we plan to fine-tune the tracking head using our stronger encoder, which we believe can provide more accurate and robust features to further enhance tracking performance.

Multi-View Pose Estimation. We evaluate our method primarily on the RealEstate10k dataset [88], following the procedure in VGGSfM [62] that involves randomly sampling 10 frames from each sequence for pose evaluation. Since some of the original YouTube links in RealEstate10k are unavailable, our evaluation is conducted on 1,756 out of the original 1,800 scenes.

Ablation Mix Dataset. For the ablation study, we replace the SA-1B dataset [26] with a mixed dataset composed of MegaDepth [30], CO3Dv2 [45], ARkitScene [5], Scannet++ [81], Scannet [10], VirtualKIITIv2 [7], BlendedMVS [79], and StaticThings3D [51]. Each dataset is equally weighted, providing coverage that is comparable to the DUSt3R training set.

[3] https://github.com/facebookresearch/vggt/issues/83

Table S1: **Quantitative results for multi-view pose estimation on the CO3Dv2 dataset.** "Ours" signifies the integration of our finetuning method. Best results in each session are highlighted in **bold**.

Methods	CO3Dv2		
	RRA@5	RTA@5	AUC@30
DUSt3R [68]	80.49	75.22	81.03
DUSt3R+Ours	**85.75**	**78.02**	**82.83**
CUT3R [65]	70.83	**64.39**	**74.10**
CUT3R+Ours	**70.89**	63.76	73.74
VGGT [61]	95.20	**84.28**	88.35
VGGT+Ours	**95.47**	84.18	**88.77**

B Additional Experiments

B.1 Multi-view Pose On CO3Dv2

We also conduct experiments on multi-view pose estimation using the CO3Dv2 dataset [45]. Following the evaluation protocol in PoseDiffusion [63], we select the first 10 frames from each sequence for evaluation. The results are presented in Table S1. Our fine-tuning improves DUSt3R by refining the geometry-based correspondence. However, the performance of CUT3R on CO3Dv2 is negatively affected, and the impact on VGGT is marginal. We suspect this is primarily because CO3Dv2 is used to train the pose head, causing it to strongly memorize the dataset.

B.2 Multi-view Feature on Feat2GS

We further provide empirical evidence to demonstrate that our approach successfully maintains multi-view consistency on Feat2GS [8] benchmark, which directly evaluates the quality of multi-view features for novel view synthesis. The results are shown in Table S2. Our method not only preserves but slightly improves multi-view performance, evidenced by the gains in PSNR and LPIPS. It is important to contextualize these numbers: performance on Feat2GS is typically concentrated within a very narrow range (e.g., PSNR often between 19.40 and 19.70). This demonstrates our method successfully improves single-view geometry while preserving the integrity of multi-view features.

Table S2: **Quantitative comparison on Feat2GS [8] benchmark.**

	Geometry			**Texture**			**All**		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
DUSt3R	19.56	0.6504	0.3181	18.06	0.6006	0.3221	19.40	0.6477	0.3700
DUSt3R_ft	19.60	0.6512	0.3181	18.05	0.6015	0.3217	19.65	0.6417	0.3669
VGGT_e	19.66	0.6558	0.3123	18.07	0.6003	0.3225	19.61	0.6510	0.3788
VGGT_e_ft	19.70	0.6561	0.3115	18.10	0.6008	0.3224	19.66	0.6514	0.3781

C Additional Visualizations

We provide additional visualizations on diverse, in-the-wild data in Figures S1, S2, and S3 to demonstrate how our fine-tuning method robustly enhances the original baseline. More visualizations can be found in the Supplementary Video, which includes fly-through sequences of the multi-view reconstruction results.

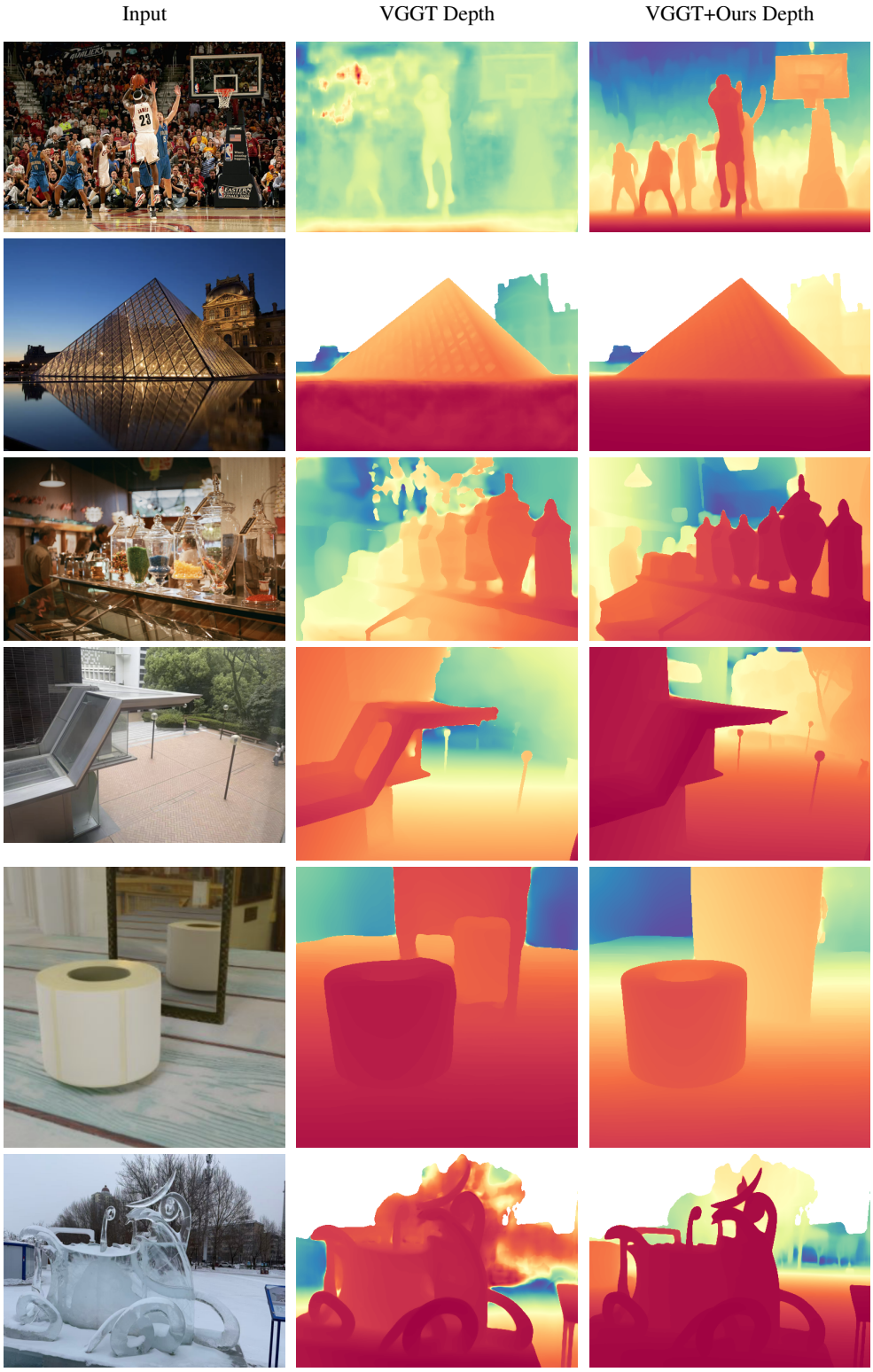


Figure S1: Additional Visualization of Depth Estimation Results: Input Images, Baseline (VGGT Depth), and Improved Method (VGGT+Ours Depth)

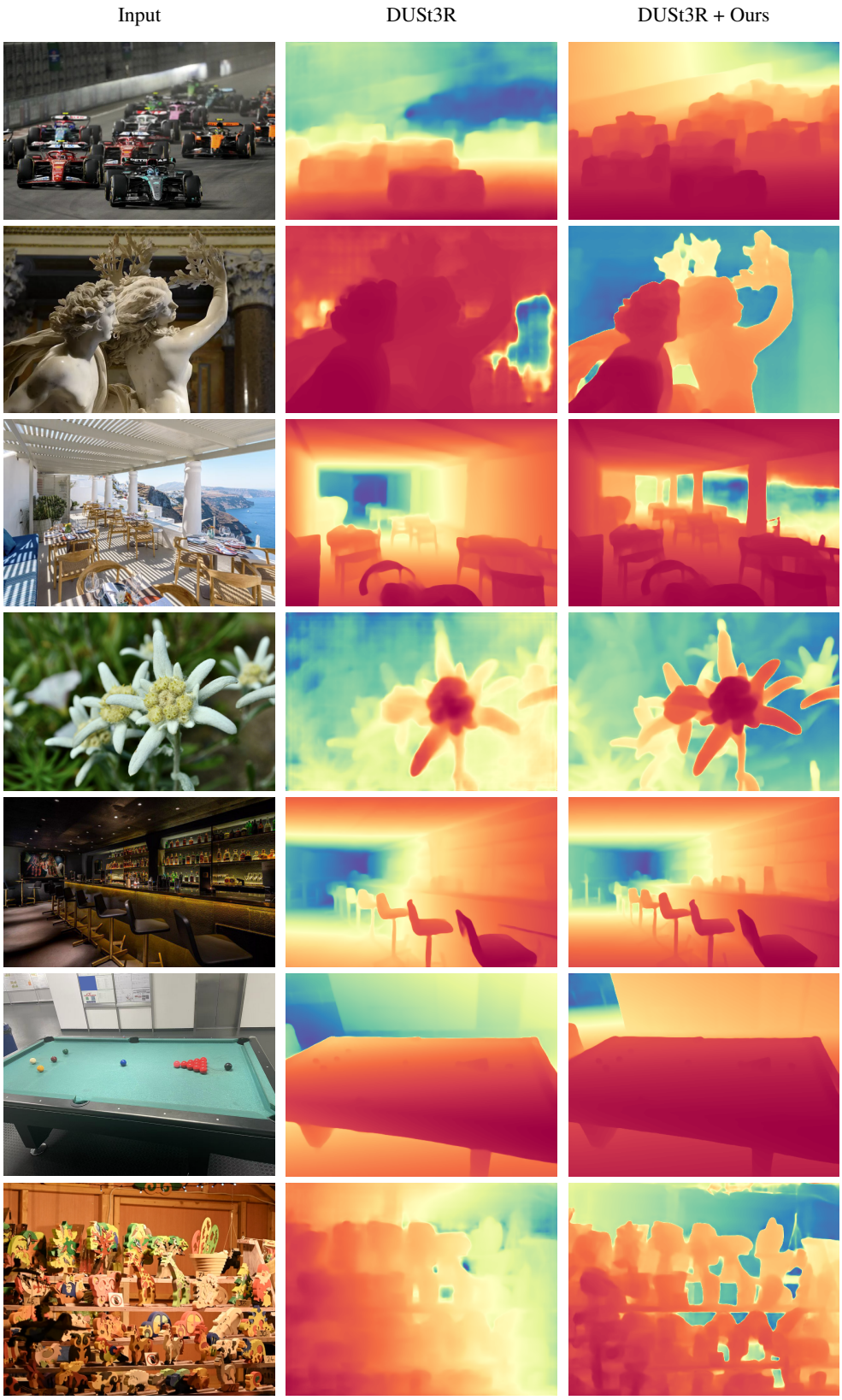


Figure S2: **Additional visualization of depth estimation results.** From left to right: input image, baseline (DUSt3R), and our improved method (DUSt3R+Ours).

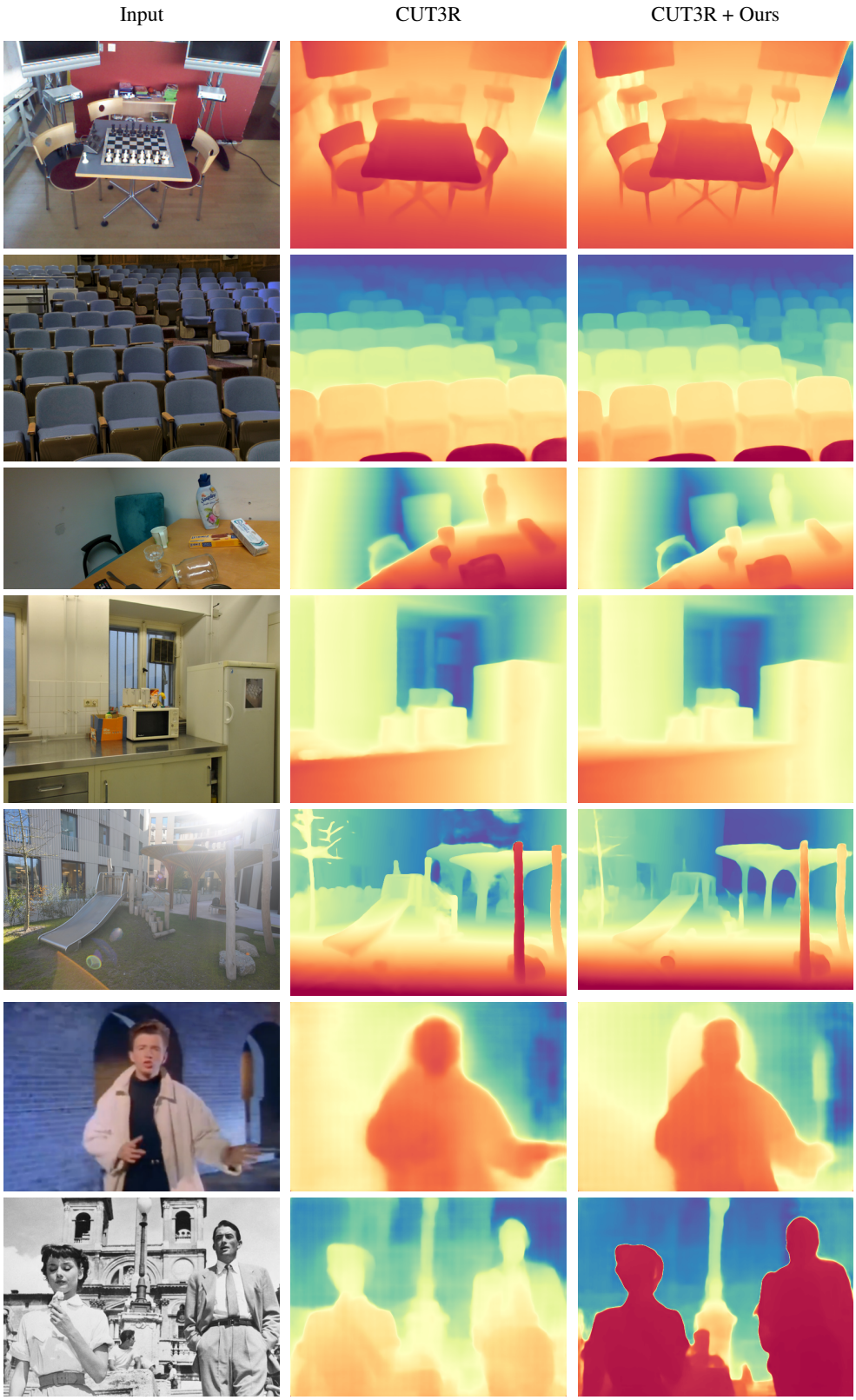


Figure S3: **Additional visualization of depth estimation results.** From left to right: input image, baseline (CUT3R), and our improved method (CUT3R+Ours).

D Proof for Long-Sequence Scale Uncertainty

Let a 3D point in the world coordinate system be

$$\mathbf{p} = \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

with a multiplicative scale uncertainty modelled as

$$\hat{\mathbf{p}} = (1+\delta)\mathbf{p}, \quad \delta \sim \mathcal{N}(0, \sigma^2),$$

where δ is a small perturbation. Under a rigid transformation characterized by rotation R and translation T, the unperturbed point in the second view is given by

$$\mathbf{p}_2 = \mathbf{R}\mathbf{p} + \mathbf{T}.$$

When the uncertainty is introduced, the perturbed second-view point becomes

$$\hat{\mathbf{p}}_2 = \mathbf{R}\hat{\mathbf{p}} + \mathbf{T} = \mathbf{R}\big[(1+\delta)\mathbf{p}\big] + \mathbf{T} = \mathbf{p}_2 + \delta\,(\mathbf{R}\mathbf{p}).$$

We define the rotated coordinates by writing

$$\mathbf{R}\mathbf{p} = \begin{bmatrix} \alpha \\ * \\ \beta \end{bmatrix},$$

where, due to the relationship $\mathbf{p}_2 = \mathbf{R}\mathbf{p} + \mathbf{T}$, the first and third components satisfy:

$$\alpha = X_2 - T_x, \quad \beta = Z_2 - T_z,$$

with $\mathbf{p}_2 \triangleq \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix}$.

Assuming a pinhole camera model with focal length f, the unperturbed horizontal image coordinate is given by

$$u = \frac{f\,X_2}{Z_2}.$$

For the perturbed coordinates we express

$$X_2^{\delta} = X_2 + \delta\,\alpha, \quad Z_2^{\delta} = Z_2 + \delta\,\beta.$$

Thus, the image coordinate under perturbation is

$$u(\delta) = \frac{f\,(X_2 + \delta\,\alpha)}{Z_2 + \delta\,\beta}.$$

Our goal is to analyze the induced projection error,

$$\Delta u \triangleq u(\delta) - u,$$

without using a Taylor expansion. We begin by forming the exact difference:

$$\Delta u = \frac{f\,(X_2 + \delta\,\alpha)}{Z_2 + \delta\,\beta} - \frac{f\,X_2}{Z_2}.$$

By combining the terms over a common denominator, we have:

$$\Delta u = f\left(\frac{(X_2 + \delta\,\alpha)Z_2 - X_2(Z_2 + \delta\,\beta)}{Z_2\,(Z_2 + \delta\,\beta)}\right).$$

Expanding the numerator yields:

$$(X_2 + \delta\,\alpha)Z_2 - X_2(Z_2 + \delta\,\beta) = X_2Z_2 + \delta\,\alpha\,Z_2 - X_2Z_2 - \delta\,X_2\,\beta = \delta\,\big(\alpha\,Z_2 - X_2\,\beta\big).$$

Thus, the error simplifies to:

$$\Delta u = \delta\,f\,\frac{\alpha\,Z_2 - X_2\,\beta}{Z_2\,(Z_2 + \delta\,\beta)}.$$

Substituting the expressions $\alpha = X_2 - T_x$ and $\beta = Z_2 - T_z$, we obtain:

$$\alpha\, Z_2 - X_2\, \beta = (X_2 - T_x)Z_2 - X_2(Z_2 - T_z) = X_2 T_z - T_x Z_2.$$

Therefore, the error becomes:

$$\Delta u = \delta\, f\, \frac{X_2 T_z - T_x Z_2}{Z_2\,(Z_2 + \delta\,(Z_2 - T_z))},$$

since $\beta = Z_2 - T_z$.

To gain further insight into the dependency on depth Z_2, let us assume that along object boundaries the ratio X_2/Z_2 remains approximately constant, i.e.,

$$X_2 \approx c\, Z_2,$$

for some constant c. Under this assumption, the numerator approximates as

$$X_2 T_z - T_x Z_2 \approx Z_2\,(c\, T_z - T_x).$$

Substituting this back, we get:

$$\Delta u \approx \delta\, f\, \frac{Z_2\,(c\, T_z - T_x)}{Z_2\,(Z_2 + \delta\,(Z_2 - T_z))} = \delta\, f\, \frac{c\, T_z - T_x}{Z_2 + \delta\,(Z_2 - T_z)}.$$

For small δ, the term $\delta\,(Z_2 - T_z)$ in the denominator is negligible compared to Z_2. That is,

$$Z_2 + \delta\,(Z_2 - T_z) \approx Z_2.$$

Thus, we arrive at the simplified expression:

$$\Delta u \approx \delta\, \frac{f\,(c\, T_z - T_x)}{Z_2}.$$

This result shows that the projection error Δu is inversely proportional to Z_2, meaning that foreground points (with small Z_2) experience larger epipolar displacements due to scale uncertainty—a phenomenon we term foreground erosion. Moreover, our analysis demonstrates that, except for the first view, the normalization process amplifies minor scale errors in the foreground; this amplification results in substantial epipolar displacement and the erosion of fine details in these regions.

E Limitations

Although our method enhances fine geometric details, its boundary accuracy remains inferior to that of MoGe [66], which produces sharper results. A mixed-teacher strategy or a more dedicated distillation design may offer further improvements. Additionally, the current model supports only 512/518 resolution, and scaling to higher resolutions remains a challenge for future work.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 2011. 3

[2] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, 2022. 8, 9

[3] Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *CVPR*, 2024. 3

[4] Daniel Barath and Chris Sweeney. Relative pose solvers using monocular depth. In *ICPR*, 2022. 3

[5] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *NeurIPS*, 2021. 23

[6] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 8

[7] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020. 23

[8] Yue Chen, Xingyu Chen, Anpei Chen, Gerard Pons-Moll, and Yuliang Xiu. Feat2gs: Probing visual foundation models with gaussian splatting. In *CVPR*, 2025. 24

[9] Jan Czarnowski, Tristan Laidlow, Ronald Clark, and Andrew J Davison. Deepfactors: Real-time probabilistic dense monocular slam. *RAL*, 2020. 3

[10] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 7, 8, 23

[11] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. *arXiv preprint arXiv:2412.08376*, 2024. 7, 8

[12] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *CVPR*, 2019. 3

[13] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. RoMa: Robust Dense Feature Matching. *CVPR*, 2024. 8

[14] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *ECCV*, 2010. 3

[15] Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *NeurIPS*, 2022. 3

[16] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015. 3

[17] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *ICCV*, 2015. 3

[18] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *CVPR*, 2020. 3

[19] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 3

[20] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 2

[21] Wonbong Jang, Philippe Weinzaepfel, Vincent Leroy, Lourdes Agapito, and Jerome Revaud. Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. *arXiv preprint arXiv:2503.17316*, 2025. 3

[22] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, 2014. 9

[23] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. 2, 3

[24] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *NeurIPS*, 2017. 6

[25] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 2, 5, 23

[26] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 6, 23

[27] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM TOG*, 2017. 8

[28] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024. 2, 3, 6, 8

[29] Wenyu Li, Sidun Liu, Peng Qiao, and Yong Dou. Mono3r: Exploiting monocular cues for geometric 3d reconstruction. *arXiv preprint arXiv:2504.13419*, 2025. 3

[30] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 23

[31] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. *arXiv preprint arXiv:2306.13643*, 2023. 3

[32] Sheng Liu, Xiaohan Nie, and Raffay Hamid. Depth-guided sparse structure-from-motion for movies and tv shows. In *CVPR*, 2022. 3

[33] Shing Yan Loo, Syamsiah Mashohor, Sai Hong Tang, and Hong Zhang. Deeprelativefusion: Dense monocular slam using single-image relative depth prediction. In *IROS*, 2021. 3

[34] Jiahao Lu, Tianyu Huang, Peng Li, Zhiyang Dou, Cheng Lin, Zhiming Cui, Zhen Dong, Sai-Kit Yeung, Wenping Wang, and Yuan Liu. Align3r: Aligned monocular depth estimation for dynamic videos. *arXiv preprint arXiv:2412.03079*, 2024. 3

[35] Ziqi Lu, Heng Yang, Danfei Xu, Boyi Li, Boris Ivanovic, Marco Pavone, and Yue Wang. Lora3d: Low-rank self-calibration of 3d geometric foundation models. *arXiv preprint arXiv:2412.07746*, 2024. 3

[36] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *ACM TOG*, 2020. 3

[37] Zeyu Ma, Zachary Teed, and Jia Deng. Multiview stereo with cascaded epipolar raft. In *ECCV*, 2022. 3

[38] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*, 2020. 3

[39] Onur Özyeşil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion*. *Acta Numerica*, 26:305–364, 2017. 3

[40] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *IROS*, 2019. 8

[41] Linfei Pan, Daniel Barath, Marc Pollefeys, and Johannes Lutz Schönberger. Global Structure-from-Motion Revisited. In *ECCV*, 2024. 2

[42] Zador Pataki, Paul-Edouard Sarlin, Johannes L. Schönberger, and Marc Pollefeys. MP-SfM: Monocular Surface Priors for Robust Structure-from-Motion. In *CVPR*, 2025. 3

[43] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *CVPR*, 2024. 2, 3

[44] Matia Pizzoli, Christian Forster, and Davide Scaramuzza. Remode: Probabilistic, monocular dense reconstruction in real time. In *ICRA*, 2014. 3

[45] 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 *ICCV*, 2021. 23, 24

[46] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, 2021. 6, 23

[47] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, pages 4938–4947, 2020. 3, 7, 8

[48] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 2, 3

[49] Thomas Schöps, Torsten Sattler, and Marc Pollefeys. BAD SLAM: Bundle adjusted direct RGB-D SLAM. In *CVPR*, 2019. 8

[50] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, pages 3260–3269, 2017. 9

[51] Philipp Schröppel, Jan Bechtold, Artemij Amiranashvili, and Thomas Brox. A benchmark and a baseline for robust multi-view depth estimation. In *3DV*, 2022. 23

[52] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pages 2930–2937, 2013. 8, 9

[53] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 3

[54] Jiuhn Song, Seonghoon Park, Honggyu An, Seokju Cho, Min-Seop Kwak, Sungjin Cho, and Seungryong Kim. Därf: boosting radiance fields from sparse inputs with monocular depth adaptation. In *NeurIPS*, 2023. 3

[55] Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. *arXiv preprint arXiv:2412.06974*, 2024. 2, 3

[56] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *CVPR*, 2017. 3

[57] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *NeurIPS*, 2021. 3

[58] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *3DV*, 2017. 8

[59] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE TPAMI*, 1991. 8

[60] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024. 2, 3, 8

[61] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, 2025. 2, 3, 4, 6, 8, 9, 24

[62] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *CVPR*, 2024. 3, 23

[63] Jianyuan Wang, Christian Rupprecht, and David Novotny. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *ICCV*, pages 9773–9783, 2023. 24

[64] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. *arXiv preprint arXiv:2501.12387*, 2025. 2, 3

[65] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state, 2025. 2, 3, 4, 6, 7, 8, 9, 24

[66] Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. *arXiv preprint arXiv:2410.19115*, 2024. 2, 3, 5, 6, 23, 29

[67] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *NeurIPS*, 2025. 6

[68] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 2, 3, 4, 6, 8, 9, 24

[69] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. *IROS*, 2020. 6, 23

[70] Yifan Wang, Xingyi He, Sida Peng, Dongli Tan, and Xiaowei Zhou. Efficient LoFTR: Semi-dense local feature matching with sparse-like speed. In *CVPR*, 2024. 8

[71] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Scalable permutation-equivariant visual geometry learning. *arXiv preprint 2507.13347*, 2025. 9

[72] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *ICCV*, 2021. 3

[73] Guangkai Xu, Yongtao Ge, Mingyu Liu, Chengxiang Fan, Kangyang Xie, Zhiyue Zhao, Hao Chen, and Chunhua Shen. What matters when repurposing diffusion models for general dense perception tasks? *arXiv preprint arXiv:2403.06090*, 2024. 3

[74] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *CVPR*, 2025. 3

[75] 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 *CVPR*, June 2025. 2

[76] 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. *arXiv preprint arXiv:2501.13928*, 2025. 3

[77] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 2, 3, 4

[78] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, 2018. 3

[79] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *CVPR*, 2020. 23

[80] 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. 8

[81] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023. 23

[82] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. LIFT: Learned Invariant Feature Transform. In *ECCV*, 2016. 3

[83] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *ICCV*, 2023. 3

[84] Yifan Yu, Shaohui Liu, Rémi Pautrat, Marc Pollefeys, and Viktor Larsson. Relative pose estimation through affine corrections of monocular depth priors. In *CVPR*, 2025. 3

[85] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. In *NeurIPS*, 2022. 3

[86] Yuheng Yuan, Qiuhong Shen, Shizun Wang, Xingyi Yang, and Xinchao Wang. Test3r: Learning to reconstruct 3d at test time. *NeurIPS*, 2025. 3

[87] 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. *CVPR*, 2025. 2, 3

[88] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM TOG*, 2018. 8, 9, 23

[89] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. In *3DV*, 2024. 3