FrozenRecon: Pose-free 3D Scene Reconstruction with Frozen Depth Models

Guangkai Xu[1]*,   Wei Yin[2]*,   Hao Chen[3],   Chunhua Shen[3,4],   Kai Cheng[1],   Feng Zhao[1]†

[1] University of Science and Technology of China   [2] DJI Technology   [3] Zhejiang University, China   [4] Ant Group

Figure 1: **Robust 3D scene reconstruction of challenging and diverse scenes.** Given a monocular video, our algorithm reconstructs the 3D scene without requiring offline-acquired camera poses. Significantly, our approach only involves dozens of parameters of each frame to optimize online. Here, the red triangulations denote the estimated camera trajectories.

Abstract

3D scene reconstruction is a long-standing vision task. Existing approaches can be categorized into geometry-based and learning-based methods. The former leverages multi-view geometry but may face catastrophic failures due to the reliance on accurate pixel correspondence across views, while the latter mitigates these issues by learning 2D or 3D representation directly. However, without a large-scale video or 3D training data, it can hardly be generalized to diverse real-world scenarios due to the presence of tens of millions or even billions of optimization parameters in the deep network.

Recently, robust monocular depth estimation models trained with large-scale datasets have been proven to possess weak 3D geometry prior, but they are insufficient for reconstruction due to the unknown camera parameters, the affine-invariant property, and inter-frame inconsistency. To address these issues, we propose a novel test-time optimization approach that can transfer the robustness of affine-invariant depth models such as LeReS to challenging diverse scenes while ensuring inter-frame consistency, with only dozens of parameters to optimize per video frame. Specifically, our approach involves freezing the pre-trained affine-invariant depth model's depth predictions, rectifying them by optimizing the unknown scale-shift values with a geometric consistency alignment module, and employing the resulting scale-consistent depth maps to robustly obtain camera poses and achieve dense scene reconstruction, even in low-texture regions. Experiments show that our method achieves state-of-the-art cross-dataset reconstruction on five zero-shot testing datasets. Code is available at: https://aim-uofa.github.io/FrozenRecon/

---

*Both authors contributed equally. † Corresponding author. Email: fzhao956@ustc.edu.cn.

1. Introduction

Dense 3D reconstruction is a fundamental vision task, which has a wide range of applications in autonomous driving [7], virtual/augmented reality [4, 37], robot navigation, medical-CAD modeling [32], *etc*. Existing geometry-based methods and learning-based methods have achieved impressive performance. Despite progress, we observe failure cases in certain real-world scenarios, such as incomplete or noisy reconstruction of low-texture scenes, tracking failures for pose estimation, limited generalization to unseen scenes, *etc*. In this work, *we aim to robustly and efficiently reconstruct diverse scenes on monocular videos, with the intrinsic camera parameters and poses jointly optimized at the same time.*

Most 3D scene reconstruction algorithms can be categorized into learning-based [20, 30, 35] and geometry-based methods [43, 2, 3, 14, 38, 17, 15, 16]. The first group of methods typically rely on a strong neural network to learn the scene geometry, which often requires large-scale and high-quality data to optimize for millions or billions of learnable parameters. Such an expensive data requirement limits applications to various scenarios. Furthermore, some methods, such as SC-DepthV3 [31], optimize the depth and poses at the same time, for which the optimization can be challenging and may become stuck in trivial solutions, partly due to the large number of optimization parameters.

In contrast, geometry-based methods find feature correspondences across views to achieve dense 3D reconstruction. Thus, fewer/no training data are needed. The drawback is that these methods can easily fail on texture-less or low-texture scenes. Moreover, planar scenes and in-place rotations also lead to degeneration in camera pose optimization. Occlusion, lighting changes, and low-texture regions can make the dense matching intractable and thus lead to incomplete reconstructions. Seeking to mitigate these limitations, we propose to exploit a pre-trained, robust monocular depth model to obtain scene geometry priors, which can largely ease pose optimization and reconstruction.

Recently, foundation models [21, 1] trained with large-scale datasets can generalize to new datasets/tasks with few samples by optimizing for a small portion of parameters (so-called adaptors). Inspired by the success, we use a pre-trained monocular depth model of strong performance such as LeReS [41], and optimize for a very sparse set of parameters for quickly rectifying the depth maps on test videos, such that scale-consistent depth maps can be attained.

Models such as LeReS [41], MiDaS [23], and DPT [22] use millions of training images to train a robust monocular depth model, which generalizes well to diverse scenes. Unfortunately, the predicted depth maps of those models are affine-invariant, *i.e.*, up to an unknown scale and shift compared against the ground-truth metric depth. As pointed out in [41], the unknown shift can cause significant distortion.

Naively fusing pre-frame prediction is prone to cause reconstruction distortion and scale misalignment. Furthermore, the unknown intrinsic camera parameters and poses are another obstacle for multi-frame reconstruction. If we have access to accurate camera parameters, we can fine tune the pre-trained model to solve the above issues, as in [17].

Instead, we freeze the monocular depth model, and on the given videos, we optimize for the global scale value, the global shift value, a local scale map and a local shift map to rectify each predicted affine-invariant depth map. Intrinsic camera parameters and camera poses are also optimized at the same time. *The number of parameters that need to be optimized online is only around 30 per frame.* This is a sharp contrast compared with learning-based approaches, *e.g.*, SC-DepthV3, where tens of millions of parameters are involved in optimization. The shift and scale parameters are optimized by supervising the photometric consistency and geometric consistency between selected keyframes. Due to the sparsity of optimization parameters and the robustness of the monocular depth model, our method works much better in terms of domain gap compared to existing deep learning methods for this task. At the same time, compared with traditional geometry-based 3D scene reconstruction methods, ours can be more robust to low-texture regions.

In order to validate the robustness of our system, we test on 5 unseen datasets: NYU [28], ScanNet [8], 7-Scenes [27], TUM [29], and KITTI [12]. Experiments show that our pipeline outperforms recent methods and achieves state-of-the-art reconstruction performance. Besides, we also perform extensive ablation studies to explore the effectiveness of components of our reconstruction pipeline. Our main contributions are summarized as follows:

- We propose a novel pipeline for dense 3D scene reconstruction by using a frozen affine-invariant depth model, and jointly optimizing a sparse set of parameters for rectifying depth maps, camera poses, and intrinsic camera parameters. Our method is robust on diverse unseen scenarios.

- Experiments on diverse datasets show the robustness of our method and verify the usefulness of each component in our method.

2. Related Work

Learning-based 3D Scene Reconstruction. Learning-based methods [20, 30, 35] rely on a neural network with a large number of learnable parameters to learn the geometry of scenes. Some approaches generate the 3D voxel volume from the 2D image features of the entire sequence [20] or local fragments [30], and estimate an implicit 3D representation from posed images. Besides, some other algorithms [43, 2, 3] attempt to predict poses and depth maps jointly.
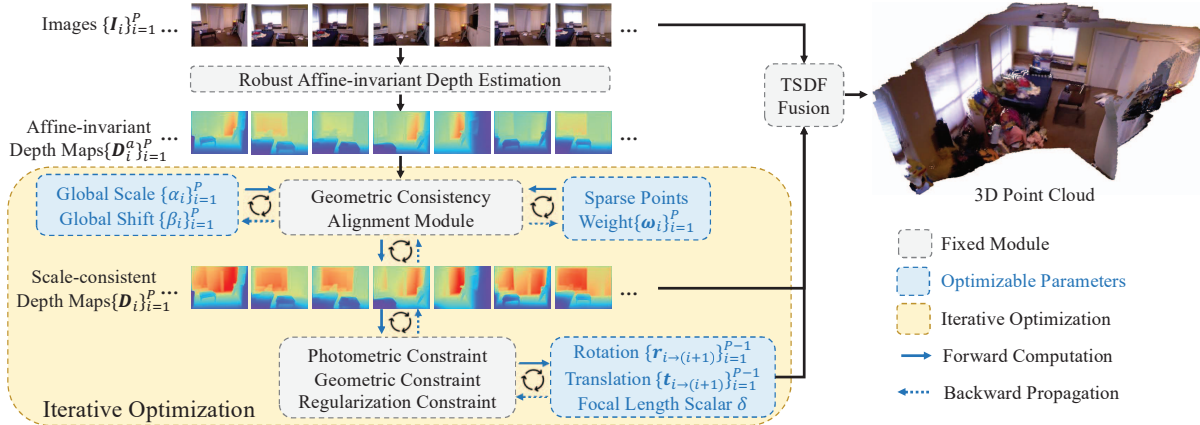
Figure 2: **The pipeline of 3D scene reconstruction.** Given a monocular video, we use a frozen, robust monocular depth estimation model to obtain the estimated depths of all frames. Then, we propose a geometric consistency alignment module, which optimizes a sparse set of parameters (*i.e.*, scale, shift, and weight factors) to achieve multi-view geometric consistent depths among all frames. The camera's intrinsic parameters and poses are also optimized simultaneously. As a result, we achieve high-quality dense 3D reconstruction with optimized depths and camera parameters.

They employ an ego-motion network and a depth network to predict relative camera poses and depth maps, and supervise the photo-metric consistency [43, 3] and geometric consistency [3].

Multi-view Geometry Based 3D Scene Reconstruction. Multi-view geometry based methods [43, 2, 3, 14, 38, 6, 13] achieve dense 3D reconstruction by finding feature correspondences across views. Traditional methods [25, 26] extract image features, establish correspondences between frames, and optimize the pose and depth iteratively, typically using bundle adjustment [34]. To achieve robust feature representation, some methods leverage plane sweeping to establish correspondences with assumed depth values. They usually construct a cost volume with the CNN-extracted features between images [13, 38, 14], and employ a ConvNet to predict depth maps after regularization. Compared with those approaches, ours is less likely to suffer from pose catastrophic failures in low-texture regions and can be more robust to diverse scenes.

Besides, visual SLAM systems [18, 19, 5, 9, 10, 11, 24, 33] also estimate the camera motion and construct the map of unknown environments. Methods of [18, 19, 5, 9] rely on the assumption of geometric or photometric consistency to estimate poses and depth through bundle adjustment. Recent works [24, 33] propose to improve the robustness and accuracy by integrating more representative clues and optimization tools from deep learning. These algorithms focus on accurate pose estimation, but may fail to build dense maps with geometry details.

Robust Monocular Depth Estimation. To achieve robust monocular depth estimation, some methods [23, 22, 41, 39]

learn the affine-invariant depth with large-scale datasets, which is more likely to be robust to unseen scenarios but up to an unknown scale and shift. Although they achieve promising robustness, the estimated affine-invariant depth needs to recover scale-shift values by globally aligning [23, 22, 41] with ground-truth depth. Some algorithms [17, 15, 16] propose to employ robust depth prediction and achieve visually consistent video depth estimation. In contrast, this work focuses on leveraging robust monocular depth models and aiming at dense 3D scene reconstruction without extra offline-obtained information.

3. Method

3.1. Overview

Aiming at 3D reconstruction from a monocular video, we propose a lightweight optimization pipeline to jointly optimize depth maps, camera poses, and intrinsic camera parameters in Figure 2. The core components of our method are the geometric consistency alignment module, optimization objectives, camera parameters initialization, and the keyframe sampling strategy.

With the sampled images $\{\mathbf{I}_i\}_{i=1}^P$, we first use LeReS [41, 40] to obtain affine-invariant depth maps $\{\mathbf{D}_i^a\}_{i=1}^P$. To avoid disastrous point cloud duplication and distortion, we leverage a geometric consistency alignment module $\mathcal{F}(\cdot, \cdot, \cdot, \cdot)$ to retrieve unknown scale and shift of affine-invariant depth and compute scale-consistent depth $\mathbf{D}_i$:

$$\mathbf{D}_i = \mathcal{F}(\mathbf{D}_i^a; \ \alpha_i, \beta_i, \boldsymbol{\omega}_i), \tag{1}$$

where α_i, β_i, and $\boldsymbol{\omega}_i$ are the optimization variables.
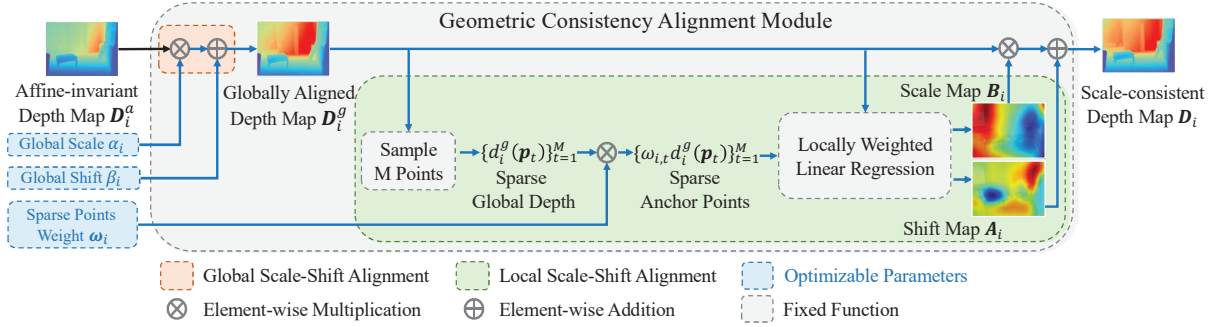
Figure 3: **The geometric consistency alignment module.** We propose to first recover the global scale and shift of depth map, then apply the locally weighted linear regression method to compute the scale map and shift map, and finally achieve scale-consistent depth.

In order to transfer the robustness of the depth model, three optimization objectives are supervised to ensure multi-frame consistency by warping from source frame i to reference frame j. With this warping process, we propose to minimize the color and depth difference between warped points and reference points, *i.e.*, photometric and geometric constraints. An additional regularization constraint is also employed to stabilize the optimization process.

During optimization, the consistency supervision relies on two conditions: 1) the warped reference image and source images must have overlaps; 2) nicely overlapped frames can be automatically selected as our keyframes. Therefore, the appropriate initialization of camera parameters and the keyframes' sampling strategy are essential.

Finally, with the optimized intrinsic camera parameters, camera poses, and scale-consistent depth maps, we can achieve accurate 3D scene reconstruction with a simple TSDF fusion [42].

3.2. Optimization

We assume a simple pinhole camera model, and use homogeneous representation for all image coordinates. By default, the scale factor in homogeneous coordinate is 1.

Geometric Consistency Alignment Module. The details of geometric consistency alignment module $\mathcal{F}(\cdot)$ are shown in Figure 3. The predicted depths $\{\mathbf{D}_i^a\}_{i=1}^{P}$ are affine-invariant. The unknown scale and shift will cause duplications and distortions if they are incorrectly estimated. To retrieve them, we perform a two-stage alignment, *i.e.*, a global scale-shift alignment, and a local scale-shift alignment. In the global alignment [41], we optimize for a global scale α_i and a global shift β_i to obtain the globally aligned depth map $\mathbf{D}_i^g$ as follows:

$$\mathbf{D}_i^g = \alpha_i \mathbf{D}_i^a + \beta_i. \tag{2}$$

In the local alignment, we compute a scale map $\mathbf{A}_i \in \mathbb{R}^{H \times W}$ and shift map $\mathbf{B}_i \in \mathbb{R}^{H \times W}$ for each frame i to re-fine the globally aligned depth. Instead of directly optimizing these two maps, we uniformly sample M sparse global depth $\{d_i^g(\mathbf{p}_t)\}_{t=1}^{M}$ from the globally aligned depth $\mathbf{D}_i^g$:

$$d_i^g(\mathbf{p}_t) = f_{\mathrm{s}}(\mathbf{D}_i^g, \mathbf{p}_{i,t}), \quad \text{for } t = 1...M,$$

where $f_{\mathrm{s}}(\cdot, \cdot)$ retrieves the depth value at a position; *e.g.*, $f_{\mathrm{s}}(\mathbf{D}_i^g, \mathbf{p}_{i,t})$ obtains $d_i^g(\mathbf{p}_t)$ from $\mathbf{D}_i^g$ at the position $\mathbf{p}_{i,t}$. Then, by multiplying sparse global depth $\{d_i^g(\mathbf{p}_t)\}_{t=1}^{M}$ with the weights $\{\omega_{i,t}\}_{t=1}^{M}$, we compute the sparse anchor points $\{\omega_{i,t} \cdot d_i^g(\mathbf{p}_t)\}_{t=1}^{M}$ and use the locally weighted linear regression method [36], *i.e.*, $f_{\mathrm{LWLR}}(\cdot, \cdot)$, to compute the local scale map $\mathbf{A}_i \in \mathbb{R}^{H \times W}$ and local shift map $\mathbf{B}_i \in \mathbb{R}^{H \times W}$. We also describe the $f_{\mathrm{LWLR}}(\cdot, \cdot)$ module in detail in the supplementary. Such indirect optimization reduces the parameters to ease optimization. The local alignment is as follows.

$$\mathbf{A}_i, \mathbf{B}_i = f_{\mathrm{LWLR}}(\mathbf{D}_i^g, \{\omega_{i,t} \cdot d_i^g(\mathbf{p}_t)\}_{t=1}^{M}),$$
$$\mathbf{D}_i = \mathbf{A}_i \odot \mathbf{D}_i^g + \mathbf{B}_i, \tag{3}$$

where $\odot$ means element-wise multiplication, and M is set to 25 in our experiments. Therefore, in this module, we only need to optimize for 27 parameters for each frame, *i.e.*, $\alpha_i, \beta_i, \omega_i$. With our geometric consistency alignment module, we can achieve scale-consistent depth.

Representation of Camera Poses and Intrinsic Camera Parameters. We propose to optimize for the relative rotation vector $\{\mathbf{r}_{i \to (i+1)}\}_{i=1}^{P-1}$ (*i.e.*, Euler angles) and relative translation vector $\{\mathbf{t}_{i \to (i+1)}\}_{i=1}^{P-1}$ between two adjacent frames. They are initialized to 0 and are transformed to 4×4 relative pose matrices $\{\mathbf{P}_{i \to (i+1)}\}_{i=1}^{P-1}$ with $\mathcal{H}(\cdot, \cdot)$. The relative pose matrices $\{\mathbf{P}_{i \to (i+1)}\}_{i=1}^{P-1}$ are transformed into camera-to-world poses matrices $\{\mathbf{P}_i\}_{i=1}^{P}$ with the product operation. Such initialization ensures the initial overlap of keyframes. For intrinsic camera parameters, we assume a simple pinhole camera model, initialize the focal length to be $f_0 = 1.2 \cdot \max(H, W)$, and learn the focal length with a learnable scalar δ. We assume that the optical center is at

the image center. Thus for camera parameters, we need to optimize for $6(P-1)+1$ variables:

$$\mathbf{P}_{i\to(i+1)} = \mathcal{H}(\mathbf{r}_{i\to(i+1)}, \mathbf{t}_{i\to(i+1)}), \quad \mathbf{P}_1 = \mathbf{E}_{4,4},$$

$$\mathbf{P}_i = \mathbf{P}_1 \prod_{k=1}^{i-1} \mathbf{P}_{k\to(k+1)}, \quad \mathbf{K} = \begin{bmatrix} \delta f_0 & 0 & W/2 \\ 0 & \delta f_0 & H/2 \\ 0 & 0 & 1 \end{bmatrix}, \quad (4)$$

where $\mathbf{E}_{4,4}$ is the 4×4 elementary matrix.

Optimization Objectives. The image coordinate $\mathbf{p}_i$ in frame i can be wrapped to frame j as follows.

$$f_s(\mathbf{D}_{i\to j}, \mathbf{p}_i)\mathbf{p}_{i\to j} = \mathbf{K}\mathbf{R}_j^\top\left[\mathbf{R}_i\mathbf{K}^{-1}f_s(\mathbf{D}_i, \mathbf{p}_i)\mathbf{p}_i + \mathbf{t}_i - \mathbf{t}_j\right],$$

$$\mathbf{p}_i, \mathbf{p}_{i\to j}, \mathbf{t}_i, \mathbf{t}_j \in \mathbb{R}^{3\times1}, \quad \mathbf{R}_i, \mathbf{R}_j, \mathbf{K} \in \mathbb{R}^{3\times3}, \quad (5)$$

where $\mathbf{K}$ is the intrinsic camera parameter. $\mathbf{R}_i$ and $\mathbf{t}_i$ are the camera-to-world rotation and translation matrices, and $\mathbf{P}_i = \begin{bmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0} & 1 \end{bmatrix}$, $\mathbf{p}_{i\to j}$ is the point warped from $\mathbf{p}_i$ of frame i to frame j, and $\mathbf{D}_{i\to j}$ is the warped depth map.

To optimize the proposed variables including $\{\alpha_i\}_{i=1}^P$, $\{\beta_i\}_{i=1}^P$, δ, $\{\mathbf{r}_{i\to(i+1)}\}_{i=1}^{P-1}$, $\{\mathbf{t}_{i\to(i+1)}\}_{i=1}^{P-1}$, and $\{\omega_i\}_{i=1}^P$, we propose to use the pixel-wise photometric and geometric constraint [2] together to ensure the consistency of color and depth along the video as follows.

$$L_{pc} = \frac{1}{|V|} \sum_{\substack{\mathbf{p} \in V \\ (i,j) \in K}} |f_s(\mathbf{I}_i, \mathbf{p}_i) - f_s(\mathbf{I}_j, \mathbf{p}_{i\to j})|, \quad (6)$$

$$L_{gc} = \frac{1}{|V|} \sum_{\substack{\mathbf{p} \in V \\ (i,j) \in K}} \frac{|f_s(\mathbf{D}_j, \mathbf{p}_{i\to j}) - f_s(\mathbf{D}_{i\to j}, \mathbf{p}_i)|}{f_s(\mathbf{D}_j, \mathbf{p}_{i\to j}) + f_s(\mathbf{D}_{i\to j}, \mathbf{p}_i)}, \quad (7)$$

where V represents all valid points successfully projected from frame i to frame j, K is the selected keyframe pairs.

Furthermore, we enforce a regularization term on the sampled sparse anchor points:

$$L_{regu} = \sum_{i=1}^N \sum_{t=1}^M |1 - \omega_{i,t}|, \quad (8)$$

The overall constraints are as follows.

$$L = \lambda_{pc}L_{pc} + \lambda_{gc}L_{gc} + \lambda_{regu}L_{regu}, \quad (9)$$

where λ_{pc}, λ_{gc}, and λ_{regu} are weights to balance them.

Optimization Details. Before we perform optimization, we first sample P frames from the video to reduce the time complexity. With these sampled frames, we perform the keyframes sampling and optimization, which consists of two stages, *i.e.*, local keyframes sampling for optimization, and global keyframes sampling for optimization.

Local keyframes sampling and optimization. In optimization, we first sample local keyframes from the k nearest

---

Algorithm 1 Optimization Algorithm

Input: N images $\{\mathbf{I}_i\}_{i=1}^N$
Output: scale-consistent depth maps $\{\mathbf{D}_i\}_{i=1}^P$, intrinsic camera parameter $\mathbf{K}$, camera-to-world camera poses $\{\mathbf{P}_i\}_{i=1}^{P-1}$
1: Sample P images $\{\mathbf{I}_i\}_{i=1}^P$ from $\{\mathbf{I}_i\}_{i=1}^N$
2: Obtain affine-invariant depth $\{\mathbf{D}_i^a\}_{i=1}^P$ from $\{\mathbf{I}_i\}_{i=1}^P$
3: Initialize the global scale $\{\alpha_i\}_{i=1}^P$, global shift $\{\beta_i\}_{i=1}^P$, focal length scalar δ, relative rotation $\{\mathbf{r}_{i\to(i+1)}\}_{i=1}^{P-1}$, relative translation $\{\mathbf{t}_{i\to(i+1)}\}_{i=1}^{P-1}$, sparse points weight $\{\omega_i\}_{i=1}^P$
4: **for** *stage* in $[Local, Global]$ **do**
5:     **for** *iter* = 1 to *iterations* **do**
6:         Compute $\{\mathbf{P}_i\}_{i=1}^{P-1}$ and $\mathbf{K}$ with Eq. (4)
7:         **if** *stage* = *Local* **then**
8:           Sample keyframe pairs $\{(i_k, j_k)\}_{k=1}^K$ with Eq. (10)
9:         **else**
10:           Sample keyframe pairs $\{(i_k, j_k)\}_{k=1}^K$ with Eq. (11)
11:         **end if**
12:         **for** keyframe pair (i, j) in $\{(i_k, j_k)\}_{k=1}^K$ **do**
13:           Compute scale-consistent depth $\mathbf{D}_i, \mathbf{D}_j$ with Eq. (3);
14:           Warp $\mathbf{p}_i$ from frame i to frame j, get warped depth $\mathbf{D}_{i\to j}$ and warped locations $\mathbf{p}_{i\to j}$ with Eq. (5)
15:           Compute optimization objectives with Eq. (9)
16:         **end for**
17:         Back-propagate and update $\{\alpha_i\}_{i=1}^P$, $\{\beta_i\}_{i=1}^P$, δ, $\{\mathbf{r}_{i\to(i+1)}\}_{i=1}^{P-1}$, $\{\mathbf{t}_{i\to(i+1)}\}_{i=1}^{P-1}$, $\{\omega_i\}_{i=1}^P$
18:     **end for**
19: **end for**
20: Compute scale-consistent depth maps $\{\mathbf{D}_i\}_{i=1}^P$ with Eq. (3), compute camera parameters $\mathbf{K}$ and $\{\mathbf{P}_i\}_{i=1}^{P-1}$ with Eq. (4)

---

frames of each reference frame. k is set to 6 in our experiments. To reduce computation time complexity, we do not use all these local keyframes with the current frame but select them based on probability. The probability for each keyframe is set as follows.

$$p_l = \begin{cases} \dfrac{1}{k}, & \text{if } k\text{-nearest neighbors}, \\ 0, & \text{otherwise}, \end{cases} \quad (10)$$

where p_l is the keyframe sample probability.

Global keyframes sampling and optimization. We further perform global keyframes sampling and optimization to improve long-range consistency. For each reference frame, all other frames are regarded as paired keyframes, but they are labeled with different sampling probabilities. We compute the relative pose between each reference frame $\mathbf{I}_i$ and any other frame $\mathbf{I}_j$. Based on the relative pose angle θ_{ij}, we set the global sampling probability p_g as follows.

$$p_g = \frac{p_l + p}{2}, p = \begin{cases} \dfrac{\theta_{ij}}{\phi^2}, & \text{if } 0 < \theta_{ij} \leq \phi \\ \dfrac{2}{\phi} - \dfrac{\theta_{ij}}{\phi^2}, & \text{if } \phi < \theta_{ij} < 2\phi, \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

Table 1: **Quantitative comparison of zero-shot 3D scene reconstruction with state-of-the-art methods.** We compare our method with seven categories of reconstruction algorithms on several video sequences of five unseen datasets: NYU [28], ScanNet [8], 7-Scenes [27], TUM [29], and KITTI [12]. Note that NeuralRecon is trained on ScanNet [8], * denotes ground-truth camera poses are given, and "Rank" means average ranking performance of each column. We evaluate the Chamfer distance C-l_1 and F-score with a threshold of 5cm. As a result, we achieve state-of-the-art rank on the five zero-shot datasets.

Method	NYU [28]		ScanNet [8]		7-Scenes [27]		TUM [29]		KITTI [12]		Rank
	C-l_1 ↓	F-score↑	C-l_1 ↓	F-score↑	C-l_1 ↓	F-score↑	C-l_1 ↓	F-score↑	C-l_1 ↓	F-score↑	
NeuralRecon* [30]	0.487	0.185	trained on ScanNet		0.235	0.219	0.435	0.241	failed on KITTI		8.125
DPSNet* [14]	0.198	0.391	0.299	0.266	0.574	0.142	0.336	0.263	**0.290**	**0.232**	4.800
BoostingDepth-DROID [36]	0.139	0.481	0.379	0.292	0.235	0.460	0.322	0.439	2.431	0.010	3.600
SC-DepthV3 [31]	0.196	0.458	0.402	0.214	0.252	0.240	0.525	0.244	4.133	0.036	5.500
CVD [17]	0.471	0.302	failed on ScanNet		0.416	0.215	0.378	0.239	5.479	0.029	7.900
RCVD [15]	0.303	0.346	0.641	0.125	0.497	0.182	0.679	0.218	58.372	0.020	8.300
GCVD [16]	0.148	0.453	0.631	0.147	0.196	0.326	0.350	0.339	2.127	0.114	4.400
COLMAP [25, 26]	0.251	0.343	0.796	0.127	0.513	0.178	0.385	0.249	107.451	0.152	7.300
DROID-SLAM [33]	0.224	0.516	0.416	0.384	0.304	**0.469**	0.285	0.433	0.686	0.155	3.200
Ours	**0.099**	**0.622**	**0.170**	**0.410**	**0.170**	0.464	**0.211**	**0.453**	0.670	0.151	**1.500**

Table 2: **Quantitative comparison of zero-shot depth estimation with state-of-the-art methods.** We evaluate the absolute relative error (AbsRel) and percentage of accurate depth pixels (δ_1) on several videos of five unseen datasets. "Rank" means average ranking performance of each column. The gray "LeReS" represents performance of affine-invariant depth without any alignment with ground-truth depth. As a result, our algorithm achieves state-of-the-art rank performance.

Method	NYU [28]		ScanNet [8]		7-Scenes [27]		TUM [29]		KITTI [12]		Rank
	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	
DPSNet* [14]	0.200	0.662	0.216	0.630	0.201	0.657	0.331	0.457	0.188	0.729	6.4
BoostingDepth-DROID [36]	0.096	0.921	0.236	0.685	0.145	0.828	0.180	0.742	**0.112**	0.860	2.7
SC-DepthV3 [31]	0.120	0.858	0.197	0.672	0.188	0.688	0.254	0.569	0.287	0.463	5.2
CVD [17]	0.167	0.837	failed on ScanNet		0.180	**0.847**	0.148	0.797	0.717	0.215	5.6
RCVD [15]	0.192	0.700	0.235	0.620	0.241	0.635	0.246	0.612	0.220	0.606	6.6
GCVD [16]	0.163	0.768	0.316	0.527	0.203	0.650	0.248	0.697	0.240	0.543	6.4
COLMAP [26, 25]	0.233	0.722	0.562	0.409	0.268	0.666	0.269	0.665	0.302	0.773	7.2
DROID-SLAM [33]	0.143	0.847	0.210	0.773	0.154	0.827	0.210	0.720	0.115	**0.914**	3.1
LeReS [41]	0.277	0.508	0.409	0.339	0.414	0.345	0.445	0.320	0.298	0.439	-
Ours	**0.092**	**0.923**	**0.127**	**0.858**	**0.135**	0.844	**0.145**	**0.799**	0.203	0.632	**1.8**

where ϕ is an angle threshold, and θ_{ij} is the relative rotation angle between the source frame $\mathbf{I}_i$ and the reference frame $\mathbf{I}_j$. p_l is the local probability with Eq. (10). The optimization is summarized in Algorithm 1.

4. Experiments

We evaluate the 3D reconstruction, scale-consistent depths, optimized camera poses, and optimized intrinsic camera parameters in experiments. More details and analyses including runtime can be found in the supplementary.

4.1. Dense 3D Scene Reconstruction

To show the robustness and accuracy of our reconstruction method, we compare it with a learning-based volumetric 3D reconstruction method (NeuralRecon [30]), a multi-view depth estimation method (DPSNet [14]), a per-frame scale-shift alignment method (BoostingDepth [36]), an unsupervised video depth estimation method (SC-DepthV3

[31]), some consistent video depth estimation methods (CVD [17], RCVD [15], GCVD [16]), a Structure-from-Motion method COLMAP [26, 25], and a robust deep visual SLAM method (DROID-SLAM [33]) on several video sequences of 5 zero-shot datasets, *i.e.*, NYU [28], ScanNet [8], 7-Scenes [27], TUM [29], KITTI [12]. Note that NeuralRecon is trained on ScanNet, while others are evaluated on unseen scenarios. NeuralRecon and DPSNet require GT poses and camera intrinsic as input, and SC-DepthV3, CVD, and DROID-SLAM require camera intrinsic. For fair comparison, we employ the estimated poses and depths of DROID-SLAM for BoostingDepth (BoostingDepth-DROID) to iteratively align the scale-shift values and filter outliers, as their paper does. Other methods optimize camera parameters on their own.

Quantitative comparisons of zero-shot 3D reconstruction and depth are shown in Table 1 and Table 2 respectively. NeuralRecon [30], DPSNet [14], SC-DepthV3 [31] need

Table 3: **Quantitative comparison of zero-shot pose estimation.** "ATE", "RPE-T", and "RPE-R" stand for the absolute trajectory error and the relative pose errors of translation and rotation, respectively. "%" represents the percentage of successfully tracked frames. Note that ORB-SLAM3 fails to track the trajectory on some low-texture regions, and can only estimate partial camera poses. Our algorithm achieves comparable performance to ORB-SLAM3 on three unseen datasets and better performance than the video depth estimation algorithms.

Method	GT Intrinsic	NYU [28] ATE↓	RPE-T↓	RPE-R↓	%↑	ScanNet [8] ATE↓	RPE-T↓	RPE-R↓	%↑	7-Scenes [27] ATE↓	RPE-T↓	RPE-R↓	%↑	TUM [29] ATE↓	RPE-T↓	RPE-R↓	%↑	KITTI [12] ATE↓	RPE-T↓	RPE-R↓	%↑
SC-DepthV3 [31]	✓	0.241	0.444	0.115	1.000	0.492	0.702	0.327	1.000	1.011	1.644	0.389	1.000	0.528	0.787	0.399	1.000	1.796	4.186	0.077	1.000
CVD [17]	✓	1.192	2.049	0.107	1.000	0.734	1.197	0.257	1.000	0.532	1.644	0.378	1.000	0.268	0.844	0.447	1.000	19.32	24.50	0.027	1.000
RCVD [15]		0.460	1.263	0.114	1.000	1.025	1.708	0.615	1.000	0.559	1.545	0.394	1.000	0.626	1.083	0.421	1.000	134.3	160.7	0.463	1.000
GCVD [16]		0.160	0.237	0.083	1.000	0.573	0.979	0.620	1.000	0.259	0.368	0.162	1.000	0.162	0.204	0.113	1.000	5.678	8.733	0.089	1.000
COLMAP [26, 25]		0.091	0.137	0.055	1.000	0.352	0.426	0.096	1.000	0.062	0.090	0.051	1.000	0.075	0.102	0.064	1.000	3.707	3.454	0.016	1.000
DROID-SLAM [33]	✓	0.050	0.079	0.032	1.000	0.230	0.230	0.052	1.000	0.050	0.072	0.072	1.000	0.044	0.077	0.079	1.000	1.491	2.302	0.005	1.000
ORB-SLAM3 [5]	✓	0.065	0.097	0.039	0.791	0.208	0.300	0.098	0.426	0.118	0.211	0.121	0.956	0.104	0.144	0.082	0.504	3.124	3.926	0.009	0.971
Ours		0.079	0.121	0.046	1.000	0.130	0.184	0.085	1.000	0.141	0.229	0.075	1.000	0.176	0.228	0.151	1.000	3.541	5.663	0.053	1.000

Table 4: **Quantitative comparison of zero-shot intrinsic camera parameter estimation.** "FOV AbsRel" represents the absolute relative error of the field of view. Note that only three algorithms can estimate the intrinsic camera parameter from the input video, and our method can achieve accurate and robust performance on four unseen datasets.

Method	NYU [28] FOV AbsRel↓	ScanNet [8] FOV AbsRel↓	7-Scenes [27] FOV AbsRel↓	TUM [29] FOV AbsRel↓	Rank
RCVD [15]	0.144	0.148	0.110	0.174	2.75
GCVD [16]	0.015	0.126	0.143	0.043	1.75
Ours	0.032	0.032	0.085	0.056	1.50

Table 5: **Runtime analysis on three representative scenes.** Our pipeline achieves state-of-the-art reconstruction but only takes around a quarter of an hour to optimize.

Method	basement_0001a	bedroom_0015	chess
NeuralRecon[30]	-	-	-
DPSNet[14]	5m 53s	45s	12m 20s
BoostingDepth-DROID[36]	35s	10s	47s
SC-DepthV3[31]	1m 47s	17s	1m 54s
CVD[17]	1h 7m 6s	12m 30s	8h 1m 34s
RCVD[15]	1h 15m 45s	10m 6s	6h 14m 54s
GCVD[16]	10m 26s	2m 37s	52m 15s
DROID-SLAM[33]	35s	16s	1m 18s
COLMAP[25, 26]	50m 44s	8m 34s	9h 40m 49s
Ours	14m 55s	6m 55s	21m 8s

to optimize millions of parameters, thus showing less robustness to unseen datasets. Rather than purely per-frame aligning with offline-obtained depth after filtering as BoostingDepth [36] does, we jointly optimize the camera parameters and depth parameters, which ensures consistency between frames and less suffers from outliers of sparse points. CVD [17] fails to reconstruct half of the ScanNet scenarios. Compared with RCVD [15], and GCVD [16], our method employs a global and a local scale-shift recovery strategy, which can rectify the affine-invariant depth to a much more accurate scale-consistent depth. Thus, we can achieve much more accurate reconstructions. The COLMAP [26, 25] struggles to estimate dense depth maps, and thus results in sparse 3D reconstruction. The robust DROID-SLAM [33] focuses on optimizing more robust poses and trajectories with the recurrent module, and spends less effort on dense reconstruction. Thus, their dense reconstruction is more likely to suffer from outliers. In contrast, our optimization is based on dense depth priors and employs a global and local optimization strategy to reduce noise, thus we in general achieve better reconstruction performance.

Quantitative comparisons of zero-shot pose estimation and intrinsic camera parameter estimation are shown in Table 3 and Table 4. Our algorithm achieves comparable performance with the traditional SLAM methods ORB-SLAM3 [5] on three datasets. However, the ORB-SLAM3

can only estimate the camera poses of partial frames due to the failure of tracking features. In contrast, our algorithm can optimize and interpolate to obtain dense trajectories. Compared to four video depth estimation methods, we can outperform them by a large margin. Note that only ours, RCVD, and GCVD can optimize the intrinsic camera parameters, while others can only employ GT camera intrinsic as input. For intrinsic camera parameter estimation, our algorithm can achieve robust and accurate performance on four unseen datasets.

Qualitative comparisons are reported in Figure. 4. NeuralRecon can only reconstruct partial meshes of unseen scenarios. The DPSNet [14] suffers from inaccurate feature correspondences, which cause noisy point clouds. BoostingDepth [36] can only filter out inaccurate points of DROID-SLAM but cannot ensure the temporal consistency, and thus fails to reconstruct consistent mesh. The SC-DepthV3 [31] suffers from the weak supervision of photometric loss and work worse on some low-texture regions. Consistent video depth estimation methods such as GCVD [16] cannot recover the shift of depth maps, which will cause distortions in 3D point clouds. DROID-SLAM [33] concentrates on accurate pose estimation but does not work
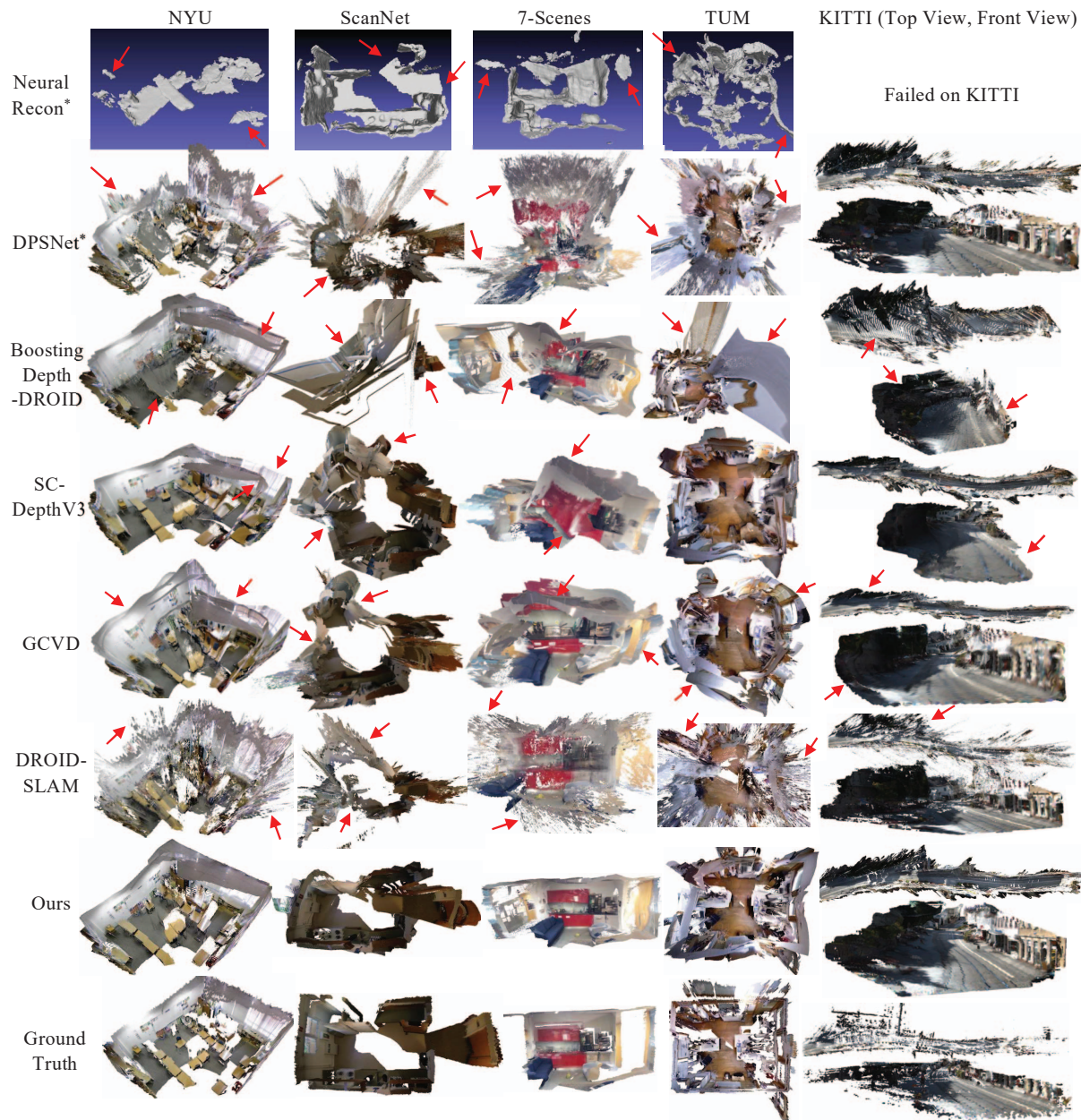
Figure 4: **Qualitative comparison of zero-shot 3D scene reconstruction from a monocular video.** We compare our method with five representative algorithms on five unseen datasets. Note that NeuralRecon is trained on ScanNet [8] and can only output uncolored mesh, and * represents the employment of ground-truth camera poses during reconstruction. These methods either suffer from inaccurate pose estimation, or predict noise depth maps (please see the red arrows). As can be seen, our approach can reconstruct better 3D scene shape without any offline-computed camera parameters.

well for dense reconstruction, and thus suffering from out-liers. In contrast, our method can achieve more robust and accurate 3D scene reconstruction, without requiring offline-acquired camera poses and camera intrinsic parameters.

Table 6: **Comparison of geometric consistency alignment mechanisms on the NYU [28] dataset.** "w/ deform." means optimizing the flexible deformation map of RCVD [15] as a replacement for our alignment module. "Baseline" means directly optimizing the per-frame pixel-wise depth map without the affine-invariant depth priors. As a result, the employment of our geometric consistency alignment module contributes significantly to the optimization.

Method	Depth		Pose			Reconstruction	
	AbsRel↓	δ_1↑	ATE↓	RPE-T↓	RPE-R↓	C-l_1↓	F-score↑
Ours	0.092	0.923	0.096	0.144	0.053	0.099	0.622
w/ deform. [15]	0.153	0.771	0.145	0.290	0.082	0.160	0.470
Baseline	0.350	0.466	0.710	1.097	0.215	0.285	0.303

Table 7: **Comparison of optimization objectives on the ScanNet [8] dataset.** The photometric constraint supervises the color consistency, the geometric constraint ensures the multi-view geometry consistency, and the regularization constraint stabilizes the convergence of optimization. As a result, the three components of our optimization objectives are all important.

Method	Depth		Pose			Reconstruction	
	AbsRel↓	δ_1↑	ATE↓	RPE-T↓	RPE-R↓	C-l_1↓	F-score↑
Ours	0.127	0.858	0.271	0.348	0.147	0.170	0.410
w/o photo.	0.219	0.636	1.722	2.331	1.039	0.775	0.169
w/o geo.	0.168	0.750	0.316	0.404	0.179	0.211	0.323
w/o regu.	0.226	0.603	0.245	0.338	0.189	0.220	0.254

For quantitative efficiency analysis, the runtime comparisons on 40 Intel Xeon Silver 4210 CPUs and an RTX 3090 Ti GPU are presented in Table 5, which includes three representative scenes with 225, 48, and 1000 images, respectively. Our pipeline achieves state-of-the-art reconstruction while only taking about a quarter of an hour to optimize, even for 1000 images. Note that only the time of predicting depth and poses without RGB-D fusion are recorded.

4.2. Ablation Studies

We carry out ablation experiments in terms of essential components of our algorithm to evaluate their effectiveness. **Geometric Consistency Alignment Module.** Table 6 shows the performance of different geometric consistency alignment modules. Previous consistent video depth methods RCVD and GCVD employ flexible deformation maps ("w/ deform.") and aim for more accurate scale-consistent depth. However, they both ignore the shift issues in affine-invariant depth, which is pretty significant in the 3D reconstruction task. By contrast, our method ("Ours") proposes a global and local scale shift alignment module for depth rectification. We also compare with a baseline method ("baseline"), which directly optimizes all frames' pixel-wise depths without the affine-invariant depth priors. All

Table 8: **Effectiveness of our two-stage keyframe sampling and optimization strategy.** The "Local" stage ensures consistency between the nearest k keyframes. The "Global" stage samples keyframes globally with different sampling probabilities according to the optimized relative poses. As a result, our algorithm achieves better performance with the coarse-to-fine optimization strategy. The experiment is conducted on NYU [28] dataset.

Stage	Depth		Pose			Reconstruction	
	AbsRel↓	δ_1 ↑	ATE↓	RPE-T↓	RPE-R↓	C-l_1 ↓	F-score↑
Local	0.111	0.884	0.111	0.179	0.068	0.091	0.593
Global	0.092	0.923	0.096	0.144	0.053	0.099	0.622

depths are initialized to 1. Our alignment module works much better than previous methods.

Optimization Objectives. To evaluate the effectiveness of each constraint employed in our method, we propose to remove them one by one, and the results are shown in Table 7. We can see that without the photometric constraint or geometric constraint, the performance will degrade a lot. When removing the regularization term, the accuracy of the pose, depth, and reconstruction will also decrease.

Effectiveness of Optimization Stages. Our optimization algorithm is composed of local keyframes optimization ("Local") and global keyframes optimization ("Global"). The local stage endures the local consistency between the nearest k keyframes, and can reconstruct roughly accurate 3D scene shapes. The global stage selects long-range keyframes to supervise global consistency. As shown in Table 8, the reconstruction performance improves gradually.

5. Conclusion

In this paper, we have presented an effective pipeline to realize 3D scene reconstruction by leveraging the robustness of affine-invariant depth estimation, freezing the depth model, and jointly optimizing dozens of depth and camera parameters for each frame. Due to the sparsity of parameters, our pipeline can transfer the robustness of seemingly weak depth geometry prior to diverse scenes. Extensive experiments show that our pipeline can achieve robust dense 3D reconstruction on challenging unseen scenes.

Acknowledgments

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Adv. Neural Inform. Process. Syst.*, 35:23716–23736, 2022. 2

[2] Jiawang Bian, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *Adv. Neural Inform. Process. Syst.*, 32:35–45, 2019. 2, 3, 5

[3] Jia-Wang Bian, Huangying Zhan, Naiyan Wang, Tat-Jun Chin, Chunhua Shen, and Ian Reid. Auto-rectify network for unsupervised indoor depth estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(12):9802–9813, 2021. 2, 3

[4] Fabio Bruno, Stefano Bruno, Giovanna De Sensi, Maria-Laura Luchi, Stefania Mancuso, and Maurizio Muzzupappa. From 3d reconstruction to virtual reality: A complete methodology for digital archaeological exhibition. *Journal of Cultural Heritage*, 11(1):42–49, 2010. 2

[5] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robotics*, 37(6):1874–1890, 2021. 3, 7

[6] Kai Cheng, Hao Chen, Wei Yin, Guangkai Xu, and Xuejin Chen. Exploiting correspondences with all-pairs correlations for multi-view depth estimation. *arXiv preprint arXiv:2205.02481*, 2022. 3

[7] Max Coenen, Franz Rottensteiner, and Christian Heipke. Precise vehicle reconstruction for autonomous driving applications. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:21–28, 2019. 2

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 5828–5839, 2017. 2, 6, 7, 8, 9

[9] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(3):611–625, 2017. 3

[10] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *Eur. Conf. Comput. Vis.*, pages 834–849. Springer, 2014. 3

[11] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. Svo: Fast semi-direct monocular visual odometry. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 15–22. IEEE, 2014. 3

[12] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *Int. J. Robotics Research*, 32(11):1231–1237, 2013. 2, 6, 7

[13] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 2821–2830, 2018. 3

[14] Sunghoon Im, Hae-Gon Jeon, Stephen Lin, and In So Kweon. Dpsnet: End-to-end deep plane sweep stereo. In *Int. Conf. Learn. Representations*, 2018. 2, 3, 6, 7

[15] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 1611–1621, 2021. 2, 3, 6, 7, 9

[16] Yao-Chih Lee, Kuan-Wei Tseng, Guan-Sheng Chen, and Chu-Song Chen. Globally consistent video depth and pose estimation with efficient test-time training. *arXiv preprint arXiv:2208.02709*, 2022. 2, 3, 6, 7

[17] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *ACM Trans. Graph.*, 39(4):71–1, 2020. 2, 3, 6, 7

[18] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE Trans. Robotics*, 31(5):1147–1163, 2015. 3

[19] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robotics*, 33(5):1255–1262, 2017. 3

[20] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *Eur. Conf. Comput. Vis.*, pages 414–431. Springer, 2020. 2

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn.*, pages 8748–8763, 2021. 2

[22] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Int. Conf. Comput. Vis.*, pages 12179–12188, 2021. 2, 3

[23] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(3):1623–1637, 2022. 2, 3

[24] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 4938–4947, 2020. 3

[25] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 4104–4113, 2016. 3, 6, 7

[26] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Eur. Conf. Comput. Vis.*, pages 501–518. Springer, 2016. 3, 6, 7

[27] 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 *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 2930–2937, 2013. 2, 6, 7

[28] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Eur. Conf. Comput. Vis.*, pages 746–760. Springer, 2012. 2, 6, 7, 9

[29] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D slam systems. In *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 573–580, 2012. 2, 6, 7

[30] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 15598–15607, 2021. 2, 6, 7

[31] Libo Sun, Jia-Wang Bian, Huangying Zhan, Wei Yin, Ian Reid, and Chunhua Shen. Sc-depthv3: Robust self-supervised monocular depth estimation for dynamic scenes. *arXiv preprint arXiv:2211.03660*, 2022. 2, 6, 7

[32] Wei Sun, Binil Starly, Jae Nam, and Andrew Darling. Bio-cad modeling and its applications in computer-aided tissue engineering. *Computer-aided design*, 37(11):1097–1114, 2005. 2

[33] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and RGB-D cameras. *Adv. Neural Inform. Process. Syst.*, 34:16558–16569, 2021. 3, 6, 7

[34] Bill Triggs, Philip McLauchlan, Richard Hartley, and Andrew Fitzgibbon. Bundle adjustment—a modern synthesis. In *International Workshop on Vision Algorithms*, pages 298–372, 1999. 3

[35] Yiming Xie, Matheus Gadelha, Fengting Yang, Xiaowei Zhou, and Huaizu Jiang. Planarrecon: Real-time 3d plane detection and reconstruction from posed monocular videos. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 6219–6228, 2022. 2

[36] Guangkai Xu, Wei Yin, Hao Chen, Chunhua Shen, Kai Cheng, Feng Wu, and Feng Zhao. Towards 3d scene reconstruction from locally scale-aligned monocular video depth. *arXiv preprint arXiv:2202.01470*, 2022. 4, 6, 7

[37] Ming-Der Yang, Chih-Fan Chao, Kai-Siang Huang, Liang-You Lu, and Yi-Ping Chen. Image-based 3d scene reconstruction and exploration in augmented reality. *Automation in Construction*, 33:48–60, 2013. 2

[38] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Eur. Conf. Comput. Vis.*, pages 767–783, 2018. 2, 3

[39] Wei Yin, Yifan Liu, and Chunhua Shen. Virtual normal: Enforcing geometric constraints for accurate and robust depth prediction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(10):7282–7295, 2021. 3

[40] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Simon Chen, Yifan Liu, and Chunhua Shen. Towards accurate reconstruction of 3d scene shape from a single monocular image. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 1–21, 2022. 3

[41] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Long Mai, Simon Chen, and Chunhua Shen. Learning to recover 3d scene shape from a single image. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 204–213, 2021. 2, 3, 4, 6

[42] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from RGB-D recon-structions. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 1802–1811, 2017. 4

[43] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 1851–1858, 2017. 2, 3