FlowMap: High-Quality Camera Poses, Intrinsics, and Depth via Gradient Descent

Cameron Smith* David Charatan* Ayush Tewari Vincent Sitzmann

MIT CSAIL

Abstract

This paper introduces FlowMap, an end-to-end differentiable method that solves for precise camera poses, camera intrinsics, and per-frame dense depth of a video sequence. Our method performs per-video gradient-descent minimization of a simple least-squares objective that compares the optical flow induced by depth, intrinsics, and poses against correspondences obtained via off-the-shelf optical flow and point tracking. Alongside the use of point tracks to encourage long-term geometric consistency, we introduce differentiable re-parameterizations of depth, intrinsics, and pose that are amenable to first-order optimization. We empirically show that camera parameters and dense depth recovered by our method enable photo-realistic novel view synthesis on 360° *trajectories using Gaussian Splatting. Our method not only far outperforms prior gradient-descent based bundle adjustment methods, but surprisingly approaches the accuracy of COLMAP, the state-of-the-art SfM method, on the downstream task of* 360° *novel view synthesis—even though our method is purely gradient-descent based, fully differentiable, and presents a complete departure from conventional SfM.*

1. Introduction

Reconstructing a 3D scene from video is one of the most fundamental problems in vision and has been studied for over five decades. Today, essentially all state-of-the-art approaches are built on top of Structure-from-Motion (SfM) methods like COLMAP [55]. These approaches extract sparse correspondences across frames, match them, discard outliers, and then optimize the correspondences' 3D positions alongside the camera parameters by minimizing reprojection error [55].

This framework has delivered excellent results which underlie many present-day vision applications, and so it is unsurprising that SfM systems have remained largely unchanged in the age of deep learning, save for deep-learning-based correspondence matching [16, 36, 53, 54].

However, conventional SfM has a major limitation: it is not differentiable with respect to its free variables (camera poses, camera intrinsics, and per-pixel depths). This means that SfM acts as an isolated pre-processing step that cannot be embedded into end-to-end deep learning pipelines. A differentiable, self-supervised SfM method would enable neural networks to be trained self-supervised on internet-scale data for a broad class of multi-view geometry problems. This would pave the way for deep-learning based 3D reconstruction and scene understanding.

In this paper, we present FlowMap, a differentiable and surprisingly simple camera and geometry estimation method whose outputs enable photorealistic novel view synthesis. FlowMap directly minimizes the difference between optical flow that is induced by a camera moving through a static 3D scene and pre-computed correspondences in the form of off-the-shelf point tracks and optical flow. Since FlowMap is end-to-end differentiable, it can naturally be embedded in any deep learning pipeline. Its loss is minimized only via gradient descent, leading to high-quality camera poses, camera intrinsics, and per-pixel depth. Unlike conventional SfM, which outputs sparse 3D points that are each constrained by several views, FlowMap outputs dense per-frame depth estimates. This is a critical advantage in downstream novel view synthesis and robotics tasks. Unlike prior attempts at gradient-based optimization of cameras and 3D geometry [2, 35, 70], we do not treat depth, intrinsics, and camera poses as free variables. Rather, we introduce differentiable feed-forward estimates of each one: depth is parameterized via a neural network, pose is parameterized as the solution to a least-squares problem involving depth and flow, and camera intrinsics are parameterized using a differentiable selection based on optical flow consistency. In other words, FlowMap solves SfM by learning the depth network's parameters; camera poses and intrinsics are computed via analytical feed-forward mod-

* Equal Contribution.
Project page: cameronosmith.github.io/flowmap

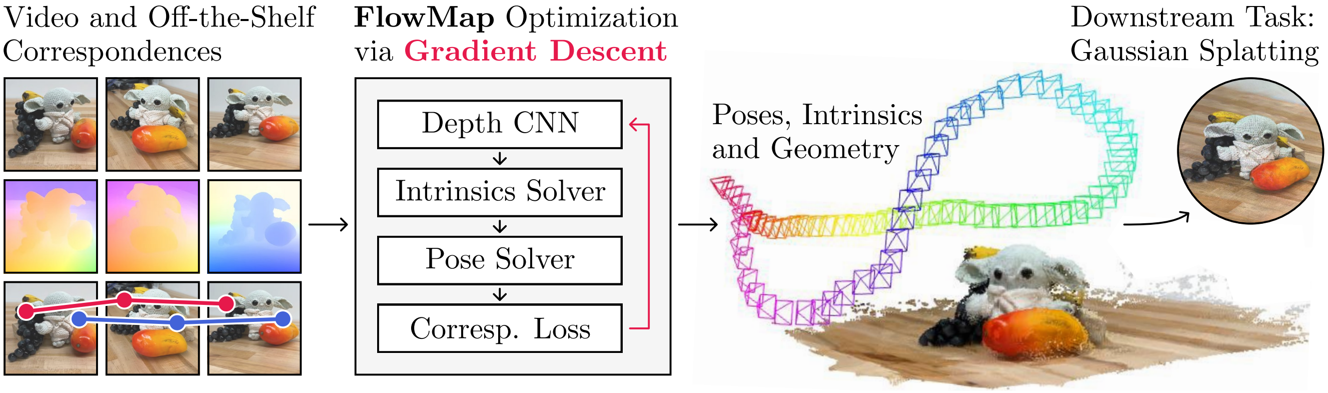

Video and Off-the-Shelf Correspondences
FlowMap Optimization via Gradient Descent
Depth CNN
Intrinsics Solver
Pose Solver
Corresp. Loss
Poses, Intrinsics and Geometry
Downstream Task: Gaussian Splatting

Figure 1. We present FlowMap, an end-to-end differentiable method that recovers poses, intrinsics, and depth maps of an input video. FlowMap is supervised only with off-the-shelf optical flow and point track correspondences, and optimized per-scene with gradient descent. Gaussian Splats obtained from FlowMap's reconstructions regularly match those obtained from COLMAP in quality.

ules without free parameters of their own. We show that this uniquely enables high-quality SfM via gradient descent while making FlowMap compatible with standard deep-learning pipelines. Unlike recent radiance-field bundle-adjustment baselines [2, 35], FlowMap does not use differentiable volume rendering, and so it is significantly faster to run, generally reconstructing an object-centric 360° scan in less than 10 minutes.

Through extensive ablation studies, we show that each of FlowMap's design choices is necessary. On popular, real-world novel view synthesis datasets (Tanks & Temples, Mip-NeRF 360, CO3D, and LLFF), we demonstrate that FlowMap enables photo-realistic novel view synthesis up to full 360° trajectories using Gaussian Splatting [29]. Gaussian Splats obtained from FlowMap reconstructions far outperform the state-of-the-art gradient-based bundle-adjustment method, NoPe-NeRF [2], and those obtained using the SLAM algorithm DROID-SLAM [64], even though both baselines require ground-truth intrinsics. Gaussian Splats obtained from FlowMap are similar to those obtained from COLMAP [55], even though FlowMap only leverages gradient descent, is fully differentiable, and represents a complete departure from conventional SfM techniques.

2. Related Work

Conventional Structure-from-Motion (SfM) and SLAM. Modern SfM methods perform offline optimization using a multi-stage process of descriptor extraction, correspondence estimation, and subsequent incremental bundle adjustment. In bundle adjustment, corresponding 2D pixels are coalesced into single 3D points, and estimated camera parameters are optimized alongside these points' 3D positions to minimize 3D-to-2D reprojection error. COLMAP [55] is the de-facto standard for accurate, offline camera parameter estimation. Meanwhile, simultaneous localization and mapping (SLAM) usually refers to real-time, online methods. These generally assume that the camera's intrinsic parameters are known. Similar to SfM, SLAM usually relies on minimizing reprojection error [6, 46, 47, 52], but some methods investigate direct minimization of a photometric error [19, 20]. While deep learning has not fundamentally transformed SfM and SLAM, it has been leveraged for correspondence prediction [12, 39, 44, 49], either via graph neural networks [53] or via particle tracking [17, 25, 80].

FlowMap is a departure from conventional SfM and SLAM techniques. While we rely on correspondence from optical flow and particle tracking, we do not coalesce sets of 2D correspondences into single 3D points. Instead, we use per-frame depth estimates as our geometry representation. Additionally, rather than relying on conventional correspondence matching and RANSAC filtering, we leverage neural point tracking [27] and optical flow estimators [63] to establish correspondence, jointly enabling dense geometry reconstruction without a seperate multi-view stereo stage. Finally, FlowMap is end-to-end differentiable and introduces feed-forward estimators of depth, poses, and intrinsics, making it compatible with other learned methods.

Deep-Learning Based SfM. Prior work has attempted to embed the full SLAM pipeline into a deep learning framework [3, 13, 14, 37, 61, 62, 66, 69, 81], usually by training black-box neural networks to directly output camera poses. However, these methods are mostly constrained to short videos of 5 to 10 frames and are not competitive with conventional SLAM and SfM for real-world 3D reconstruction. Bowen et al. [4] elegantly leverage optical flow supervision for self-supervised monocular depth prediction. More recently, DROID-SLAM [64] has yielded high-quality camera poses and depth. However, it requires known intrinsics, is trained fully supervised with ground-truth camera poses, and fails to approach COLMAP on in-the-wild per-

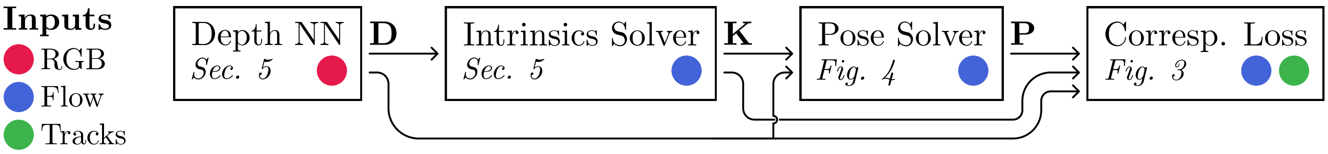


Figure 2. **A FlowMap Forward Pass.** Given RGB frames (red), optical flow (blue) and point tracks (green), FlowMap computes dense depth $\mathbf{D}$, camera poses $\mathbf{P}$, and intrinsics $\mathbf{K}$ in each forward pass. We obtain depth via a CNN (Sec. 4) and implement differentiable, feed-forward solvers for intrinsics and poses (Sec. 4, Fig.4). Colored dots indicate which block receives which inputs. FlowMap's only free parameters are the weights of a depth NN and a small correspondence confidence MLP. These parameters are optimized for each video separately by minimizing a camera-induced flow loss (Fig. 3) via gradient descent, though fully feed-forward operation is possible.

formance and robustness. Concurrent work to FlowMap explores an end-to-end differentiable, point-tracking-based SfM framework [67]. Unlike FlowMap, this method is fully supervised with camera poses, point clouds, and intrinsics; requires large-scale, multi-stage training; solves only for sparse depth; and is built around the philosophy of making each part of the conventional SfM pipeline differentiable. Another concurrent method [5] produces impressive neural SfM results by alternating between mapping and relocalization, though an initial depth estimate for one frame is required to bootstrap this process. Our method is a complete departure from the conventional SfM pipeline—it does not require a training set of known intrinsics, ground-truth poses, or 3D points, and it provides quality gradients for dense depth, poses, and intrinsics. Critically, FlowMap is among the first gradient-descent based methods to approach the performance of conventional SfM on the novel view synthesis task. Zhang et al. [79] and Kopf ef al. [32] demonstrate gradient-descent based optimization of camera parameters with a similar flow-based reprojection supervision, with a focus on dynamic scenes. However, these methods optimize camera parameters as free variables and depend on pre-trained monocular depth estimators. In constrast, our feed-forward parameterization uniquely enables gradients for large-scale training and we demonstrate that our gradients can be used to *train* a depth estimator.

Novel View Synthesis via Differentiable Rendering. Advances in differentiable rendering have enabled photo-realistic novel view synthesis and fine-grained geometry reconstruction using camera poses and intrinsics obtained via SfM [34, 43, 45, 48, 56, 57]. 3D Gaussian Splatting [29] goes further, directly leveraging the 3D points provided by SfM as an initialization. It follows previous methods like [15], which used 3D geometry from depth to supervise neural radiance field (NeRF) reconstructions. We show that when initializing Gaussian Splatting with poses, intrinsics, and 3D points from FlowMap, we generally perform on par with conventional SfM and sometimes even outperform it.

Camera Pose Optimization via Differentiable Rendering. A recent line of work in bundle-adjusting radiance fields [2, 9, 10, 21, 22, 26, 28, 35, 71–73, 76, 78] attempts to jointly optimize unknown camera poses and radiance fields. Several of these methods [24, 26, 75] additionally solve for camera intrinsic parameters. However, these methods only succeed when given forward-facing scenes or roughly correct pose initializations. More recent work incorporates optical flow and monocular depth priors [2, 38, 41] but requires known intrinsics and only works robustly on forward-facing scenes. Concurrent work [22] accelerates optimization compared to earlier NeRF-based approaches. Unlike ours, this approach requires known intrinsics and a pre-trained monocular depth estimator, and minimizes a volume-rendering-based photometric loss instead of the proposed correspondence-based approach. Further concurrent work proposes real-time SLAM via gradient descent on 3D Gaussians [40], but requires known intrinsics and does not show robustness on a variety of real-world scenes. In contrast, our method is robust and easily succeeds on object-centric scenes where the camera trajectory covers a full 360° of rotation, yielding photo-realistic novel view synthesis when combined with Gaussian Splatting.

Learning Priors over Optimization of NeRF and Poses. Our method is inspired by recent methods which learn priors over pose estimation and 3D radiance fields [8, 23, 33, 58]. However, these approaches require known camera intrinsics, are constrained to scenes with simple motion, and do not approach the accuracy of conventional SfM. Like our method, FlowCam [58] uses a pose-induced flow loss and a least-squares solver for camera pose. However, our method has several key differences: we estimate camera intrinsics, enabling optimization on any raw video; we replace 3D rendering with a simple depth estimator, which reduces training costs and allows us to reuse pre-trained depth estimators; and we introduce point tracks for supervision to improve global consistency and reduce drift. Unlike FlowMap, FlowCam did not approach conventional SfM's accuracy on real sequences. We demonstrate that optimizing the pose-induced flow objective on a single scene, akin to a test-time optimization, yields pose and geometry estimates which, for the first time, approach COLMAP on full 360° sequences.

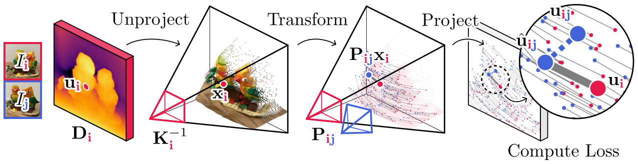


Figure 3. **Camera-Induced Flow Loss.** To use a known correspondence $\mathbf{u}_{ij}$ to compute a loss $\mathcal{L}$, we unproject $\mathbf{u}_i$ using the corresponding depth map $\mathbf{D}_i$ and camera intrinsics $\mathbf{K}_i$, transform the resulting point $\mathbf{x}_i$ via the relative pose $\mathbf{P}_{ij}$, reproject the transformed point to yield $\hat{\mathbf{u}}_{ij}$, and finally compute $\mathcal{L} = \|\hat{\mathbf{u}}_{ij} - \mathbf{u}_{ij}\|$.

3. Supervision via Induced Scene Flow

Given a video sequence, our goal is to supervise per-frame estimates of depth, intrinsics, and pose using known correspondences. Our method hinges upon the fact that a camera moving through a static scene induces optical flow in image space. Such optical flow can be computed differentiably from any two images' estimated depths, intrinsics, and relative pose to yield a set of implied pixel-wise correspondences. These correspondences can then by compared to their known counterparts to yield supervision on the underlying estimates.

Consider a 2D pixel at coordinate $\mathbf{u}_i \in \mathbb{R}^2$ in frame i of the video sequence. Using frame i's estimated depth $\mathbf{D}_i$ and intrinsics $\mathbf{K}_i$, we can compute the pixel's 3D location $\mathbf{x}_i \in \mathbb{R}^3$. Then, using the estimated relative pose $\mathbf{P}_{ij}$ between frames i and j, we can transform this location into frame j's camera space. Finally, we can project the resulting point $\mathbf{P}_{ij}\mathbf{x}_i$ onto frame j's image plane to yield an implied correspondence $\hat{\mathbf{u}}_{ij}$. This correspondence can be compared to the known correspondence $\mathbf{u}_{ij}$ to yield a loss $\mathcal{L}$, as illustrated in Fig. 3.

$$\mathcal{L} = \|\hat{\mathbf{u}}_{ij} - \mathbf{u}_{ij}\| \tag{1}$$

Supervision via Dense Optical Flow and Sparse Point Tracks. Our known correspondences are derived from two sources: dense optical flow between adjacent frames and sparse point tracks which span longer windows. Frame-to-frame optical flow ensures that depth is densely supervised, while point tracks minimize drift over time. We compute correspondences from optical flow $\mathbf{F}_{ij}$ via $\mathbf{u}_{ij} = \mathbf{u}_i + \mathbf{F}_{ij}[\mathbf{u}_i]$. Meanwhile, given a query point $\mathbf{u}_i$, an off-the-shelf point tracker directly provides a correspondence $\mathbf{u}_{ij}$ for any frame j where one exists.

Baseline: Pose, Depth and Intrinsics as Free Variables. Assuming one uses standard gradient descent optimization, one must decide how to parameterize the estimated depths, intrinsics, and poses. The simplest choice is to parameterize them as free variables, i.e., to define learnable per-camera intrinsics and extrinsics alongside per-pixel depths. However, this approach empirically fails to converge to good poses and geometry, as shown in Sec. 7.

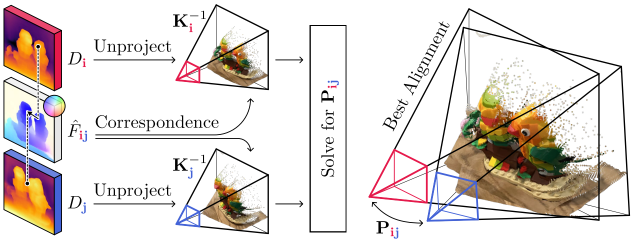


Figure 4. We solve for the relative poses between consecutive frames using their depth maps, camera intrinsics, and optical flow. To do so, we first unproject their depth maps, then solve for the pose that best aligns the resulting point clouds.

4. Parameterizing Depth, Pose, and Intrinsics

In this section, we present FlowMap's feed-forward reparameterization of depth, pose, and camera intrinsics, which uniquely enables high-quality results when using gradient descent. Later, in Sec. 7, we ablate these parameterizations to demonstrate that they lead to dramatic improvements in accuracy.

Depth Network. If each pixel's depth were optimized freely, two identical or very similar image patches could map to entirely different depths. We instead parameterize depth as a neural network that maps an RGB frame to the corresponding per-pixel depth. This ensures that similar patches have similar depths, allowing FlowMap to integrate geometry cues across frames: if a patch receives a depth gradient from one frame, the weights of the depth network are updated, and hence the depths of all similar video frame patches are also updated. As a result, FlowMap can provide high-quality depths even for patches which are poorly constrained due to errors in the input flows and point tracks, imperceptibly small motion, or degenerate (rotation-only) motion.

Pose as a Function of Depth, Intrinsics and Optical Flow. Suppose that for two consecutive frames, optical flow, per-pixel depths, and camera intrinsics are known. In this case, the relative pose between these frames can be computed differentiably in closed form. Following the approach proposed in FlowCam [58], we solve for the relative pose that best aligns each consecutive pair of un-projected depth maps. We then compose the resulting relative poses to produce absolute poses in a common coordinate system.

More formally, we cast depth map alignment as an orthogonal Procrustes problem, allowing us to draw upon this problem's differentiable, closed-form solution [11]. We begin by unprojecting the depth maps $\mathbf{D}_i$ and $\mathbf{D}_j$ using their respective intrinsics $\mathbf{K}_i$ and $\mathbf{K}_j$ to generate two point clouds $\mathbf{X}_i$ and $\mathbf{X}_j$. Next, because the Procrustes formulation requires correspondence between points, we use the known optical flow between frames i and j to match points in $\mathbf{X}_i$ and $\mathbf{X}_j$. This yields $\mathbf{X}_i^{\leftrightarrow}$ and $\mathbf{X}_j^{\leftrightarrow}$, two filtered point clouds for which a one-to-one correspondence exists. The

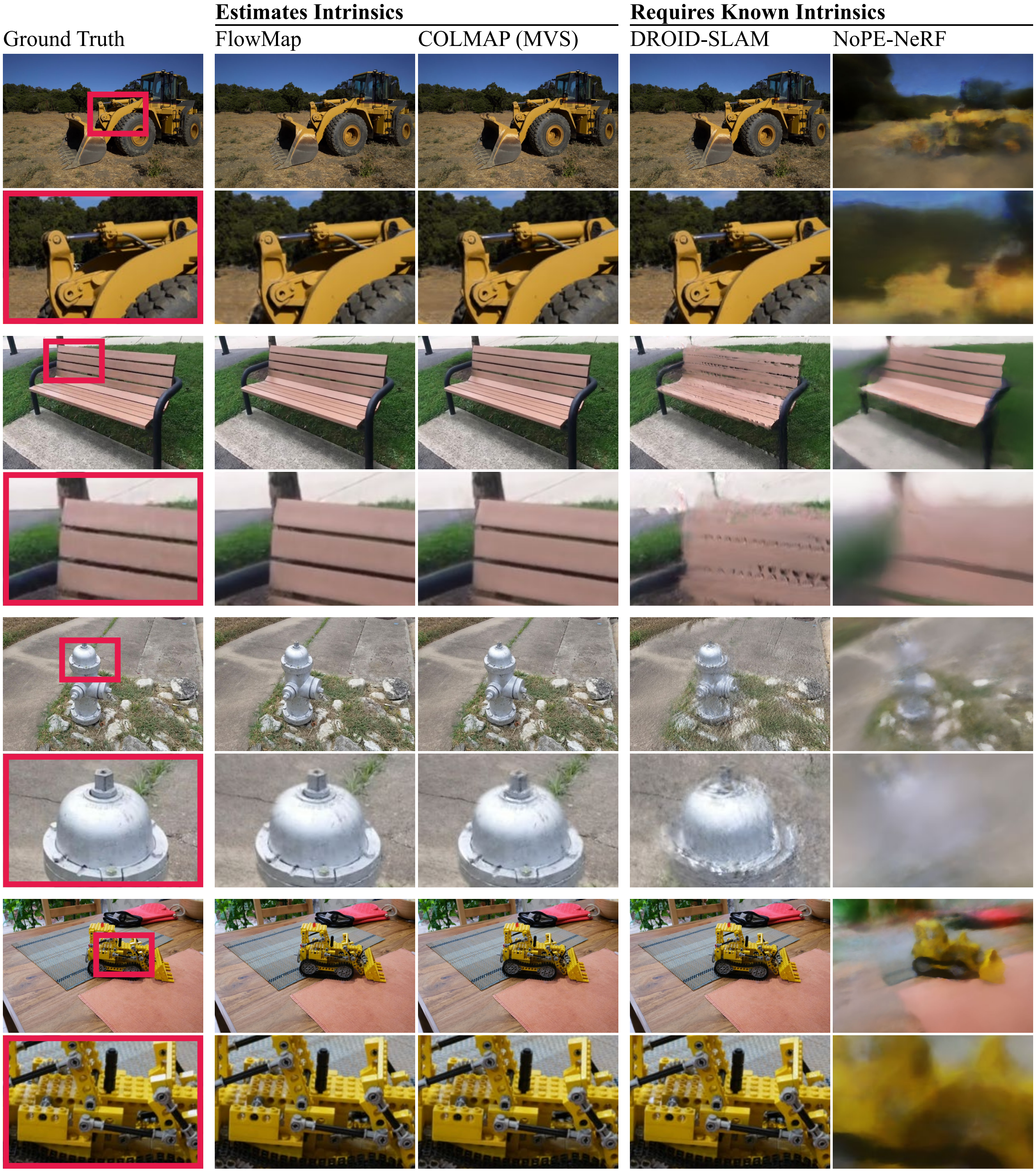


Figure 5. **View Synthesis.** FlowMap's camera parameters and geometry produce similar 3D Gaussian Splatting results to COLMAP.

Procrustes formulation seeks the rigid transformation that minimizes the total distance between the matched points:

$$\mathbf{P}_{ij} = \underset{\mathbf{P}\in\mathrm{SE}(3)}{} \|\mathcal{W}^{1/2}(\mathbf{X}_j^{\leftrightarrow} - \mathbf{P}\mathbf{X}_i^{\leftrightarrow})\|_2^2 \tag{2}$$

The diagonal matrix $\mathcal{W}$ contains correspondence weights that can down-weight correspondences that are faulty due to occlusion or imprecise flow. This weighted least-squares problem can be solved in closed form via a single singular value decomposition [11, 58] which is both cheap and fully differentiable. We further follow FlowCam [58] and predict

	MipNeRF 360 (3 scenes)					LLFF (7 scenes)				
Method	PSNR ↑	SSIM ↑	LPIPS ↓	Time (min.) ↓	ATE	PSNR ↑	SSIM ↑	LPIPS ↓	Time (min.) ↓	ATE
FlowMap	29.84	0.916	0.073	19.8	0.00055	27.23	0.849	0.079	7.5	0.00209
COLMAP	29.95	0.928	0.074	4.8	N/A	25.73	0.851	0.098	1.1	N/A
COLMAP (MVS)	31.03	0.938	0.060	42.5	N/A	27.99	0.867	0.072	13.4	N/A
DROID-SLAM*	29.83	0.913	0.066	0.6	0.00017	26.21	0.818	0.094	0.3	0.00074
NoPE-NeRF*	13.60	0.377	0.750	1913.1	0.04429	17.35	0.490	0.591	1804.0	0.03920

	Tanks & Temples (14 scenes)					CO3D (2 scenes)				
Method	PSNR ↑	SSIM ↑	LPIPS ↓	Time (min.) ↓	ATE	PSNR ↑	SSIM ↑	LPIPS ↓	Time (min.) ↓	ATE
FlowMap	27.00	0.854	0.101	22.3	0.00124	31.11	0.896	0.064	22.1	0.01589
COLMAP	26.74	0.848	0.130	5.5	N/A	25.17	0.750	0.190	12.6	N/A
COLMAP (MVS)	27.43	0.863	0.097	51.4	N/A	25.35	0.762	0.175	52.0	N/A
DROID-SLAM*	25.70	0.824	0.133	0.8	0.00122	25.97	0.790	0.139	0.8	0.01728
NoPE-NeRF*	13.38	0.449	0.706	2432.9	0.03709	14.97	0.400	0.770	2604.9	0.03648

Table 1. Camera parameter and geometry intializations from FlowMap produce 3D Gaussian reconstruction results that far outperform prior gradient-based baselines and are generally on par with those produced by COLMAP. Methods marked with an asterisk require ground-truth intrinsics. We report ATE with respect to COLMAP's pose estimates for reference, since no ground-truth trajectories exist for common view synthesis datasets. We exclude scenes where COLMAP or FlowMap fail entirely; each fails on 4 scenes. See the supplementary document for more details.

these weights by concatenating corresponding per-pixel features and feeding them into a small MLP. This MLP's parameters are the only other free variables of our model. For an overview of the depth map alignment process, see Fig. 4.

Camera Focal Length as a Function of Depth and Optical Flow. We solve for camera intrinsics by considering a set of reasonable candidates $\mathbf{K}_k$, then softly selecting among them. For each candidate, we use our pose solver Eq. 2 to compute a corresponding set of poses, then use the camera-induced flow loss Eq. 1 to compute the loss $\mathcal{L}_k$ implied by $\mathbf{K}_k$ and these poses. Finally, we compute the resulting intrinsics $\mathbf{K}$ via a softmin-weighted sum of the candidates:

$$\mathbf{K} = \sum_k w_k \mathbf{K}_k \qquad w_k = \frac{\exp(-\mathcal{L}_k)}{\sum_l \exp(-\mathcal{L}_l)} \tag{3}$$

To make this approach computationally efficient, we make several simplifying assumptions. First, we assume that the intrinsics can be represented via a single $\mathbf{K}$ that is shared across frames. Second, we assume that $\mathbf{K}$ can be modeled via a single focal length with a principal point fixed at the image center. Finally, we only compute the soft selection losses on the first two frames of the sequence.

Depth as the Only Free Variable in SfM. FlowMap offers a surprising insight: Given correspondence, SfM can be formulated as solving for per-frame depth maps. FlowMap yields poses and intrinsics in a parameter-free, differentiable forward pass when given correspondences and depths. This means that better initializations of FlowMap's depth estimator (e.g., from pre-training) will yield more accurate camera parameters (see Fig. 8).

5. Implementation and Optimization Details

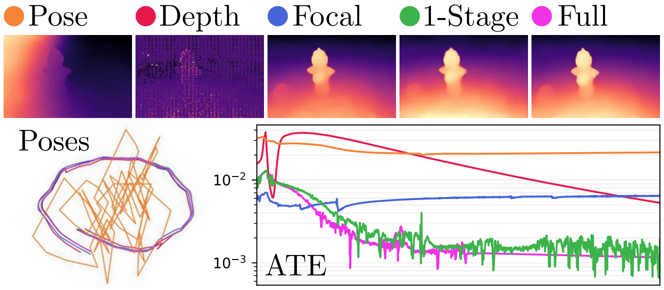
FlowMap is optimized on each specific scene, achieving convergence between 500 and 5,000 steps using the Adam [30] optimizer. Though per-scene optimization is key to achieving high accuracy, we find that exploiting FlowMap's feed-forward nature for pre-training yields an initialization that leads to improved convergence and accuracy, as shown in Fig. 8. During pre-training, in order to minimize the time spent computing correspondences, we do not use point tracks and use GMFlow [74] to compute optical flow instead of RAFT.

Focal Length Regression. While our soft selection approach robustly yields near-correct focal lengths, its performance is slightly worse compared to well-initialized direct regression. We therefore switch to focal length regression after 1,000 steps, using our softly selected focal length as initialization.

6. Results

We benchmark FlowMap via the downstream task of 3D Gaussian reconstruction [29]. This allows us to measure the quality of the camera parameters and geometry (depth maps) it outputs *without having access to ground-truth scene geometry and camera parameters*.

Baselines. We benchmark FlowMap against several baselines. First, we evaluate against COLMAP [55], the state-of-the-art structure-from-motion (SfM) method. Given a collection of images, COLMAP outputs per-image camera poses and intrinsics alongside a sparse 3D point cloud of the underlying scene. 3D Gaussian Splatting, which was designed around COLMAP's SfM outputs, is initialized using this point cloud. Second, we evaluate



Method	PSNR ↑	SSIM ↑	LPIPS ↓
● FlowMap	27.70	0.863	0.089
● Single Stage	26.66	0.842	0.112
● Expl. Focal Length	25.15	0.788	0.141
● Expl. Depth	8.84	0.168	0.684
● Expl. Pose	16.00	0.533	0.495
No Point Tracks	25.83	0.822	0.122

Figure 6. **Ablations.** We ablate the proposed feed-forward re-parameterizations of depth, pose, and intrinsics across all datasets. We find that these reparameterizations are not only critical for high-quality downstream 3D Gaussian Splatting, but also lead to dramatically accelerated convergence, where FlowMap generally converges to high quality poses within a fraction of the optimization steps required for the ablated variants. We further find that point tracks lead to a significant boost over optical flow alone (right). See the supplemental document for more ablations.

against COLMAP multi-view stereo (MVS), which enhances COLMAP's output with a much denser 3D point cloud. When initialized using this denser point cloud, 3D Gaussian Splatting produces slightly better results. However, note that COLMAP MVS is rarely used in practice because it can be prohibitively time-consuming to run. Third, we evaluate against DROID-SLAM, a neural SLAM system trained on a synthetic dataset of posed video trajectories. Finally, we evaluate against NoPE-NeRF, an method that jointly optimizes a neural radiance field and unknown camera poses. Note that unlike FlowMap and COLMAP, both DROID-SLAM and NoPE-NeRF require camera intrinsics as input.

Datasets. We analyze FlowMap on four standard novel view synthesis datasets: MipNeRF-360 [1], Tanks & Temples [31], LLFF [42], and CO3D [51]. Because FlowMap runs on video sequences, we restrict these datasets to just the video-like sequences they provide.

Methodology. We run FlowMap and the baselines using images that have been rescaled to a resolution of about 700,000 pixels. We then optimize 3D Gaussian scenes for all methods except NoPE-NeRF, since it provides its own NeRF renderings. We use 90% of the available views for training and 10% for testing. During 3D Gaussian fitting, we follow the common [60] practice of fine-tuning the initial camera poses and intrinsics. Such refinement is beneficial because the camera poses produced by SfM algorithms like COLMAP are generally not pixel-perfect [35, 50]. We use the 3D points provided by COLMAP, DROID-SLAM, and FlowMap as input to 3D Gaussian Splatting. For FlowMap, we combine the output depth maps, poses, and intrinsics to yield one point per depth map pixel.

6.1. Novel View Synthesis Results

Tab. 1 reports rendering quality metrics (PSNR, SSIM, and LPIPS) on the held-out test views, and Fig. 5 shows qualitative results. Qualitatively, FlowMap facilitates high-quality 3D reconstructions with sharp details. Quantitatively, FlowMap performs slightly better than COLMAP SfM and significantly outperforms DROID-SLAM and NoPE-NeRF. Only COLMAP MVS slightly exceeds FlowMap in terms of reconstruction quality. As noted previously, COLMAP MVS is rarely used for 3D Gaussian Splatting, since it is very time-consuming to run on high-resolution images.

6.2. Camera Parameter Estimation Results

Since the datasets we use do not provide ground-truth camera parameters, they cannot be used to directly evaluate camera parameter estimates. Instead, Tab. 1 reports the average trajectory error (ATE) of FlowMap, DROID-SLAM, and NoPe-NeRF with respect to COLMAP. Since COLMAP's poses are not perfect [50], this comparison is not to be understood as a benchmark, but rather as an indication of how close these methods' outputs are to COLMAP's state-of-the-art estimates. We find that DROID-SLAM and FlowMap both recover poses that are close to COLMAP's, while NoPE-NeRF's estimated poses are far off. When computing ATEs, we normalize all trajectories such that $\mathrm{tr}(XX^T) = 1$, where X is an n-by-3 matrix of camera positions.

Fig. 9 plots trajectories recovered by FlowMap against those recovered by COLMAP, showing that they are often nearly identical. Fig. 7 shows point clouds derived from FlowMap's estimated depth maps and camera parameters, illustrating that FlowMap recovers well-aligned scene geometry.

7. Ablations and Analysis

We perform ablations to answer the following questions:

- Question 1: Are FlowMap's reparameterizations of depth, pose, and intrinsics necessary, or do free variables perform equally well?

- Question 2: Are point tracks critical to FlowMap's performance?

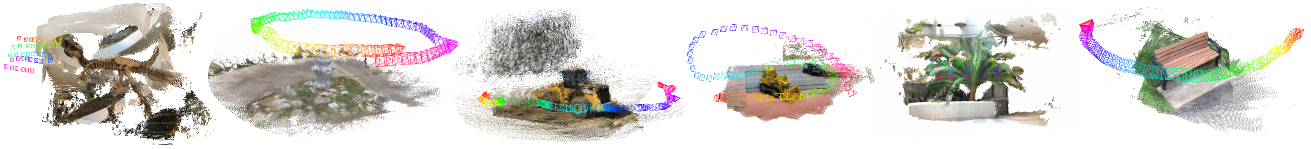


Figure 7. **Point Clouds Reconstructed by FlowMap.** Unprojecting FlowMap depths using FlowMap's intrinsics and poses yields dense and consistent point clouds.

- Question 3: Does self-supervised pre-training of the depth estimation and correspondence weight neural networks improve performance?

Parameterizations of Depth, Pose, and Camera Intrinsics (Q1) We compare the reparameterizations described in Sec. 4 to direct, free-variable optimization of pose, depth, and intrinsics. Fig. 6 shows qualitative results and quantitative results averaged across 33 scenes. We find that free-variable variants of FlowMap produce significantly worse reconstruction results and converge much more slowly, confirming that FlowMap's reparameterizations are crucial.

It is worth noting that often, explicitly optimizing a focal length produces high-quality results, as indicated by the relatively high performance of the "Expl. Focal Length" ablation. In fact, given a good initialization, direct focal length regression produces slightly better results than the proposed focal length reparameterization alone on about 80 percent of scenes. However, on about 20 percent of scenes, this approach falls into a local minimum and reconstruction fails catastrophically. This justifies the approach FlowMap uses, where the first 1,000 optimization steps use a reparameterized focal length, which is then used to initialize an explicit focal length used for another 1,000 optimization steps.

We further highlight that FlowMap's reparameterizations are necessary to estimate poses and intrinsics in a single forward pass, which is crucial for the generalizable (pre-training) setting explored in Q3.

Point Tracking (Q2) While optical flow is only computed between adjacent frames, point track estimators can accurately track points across many frames. In Fig. 6, we show that FlowMap's novel view synthesis performance drops moderately when point tracks are disabled.

Pre-training Depth and Correspondence Networks (Q3) Since FlowMap is differentiable and provides gradients for any depth-estimating neural network, it is compatible with both randomly initialized neural networks and pre-trained priors. Learned priors can come from optimization on many scenes, from existing depth estimation models, or from a combination of the two. In practice, starting with a pre-trained prior leads to significantly faster convergence, as illustrated in Fig. 8. Note that pre-training and generalization are uniquely enabled by the proposed feed-forward reparameterizations of depth, focal length, and poses.

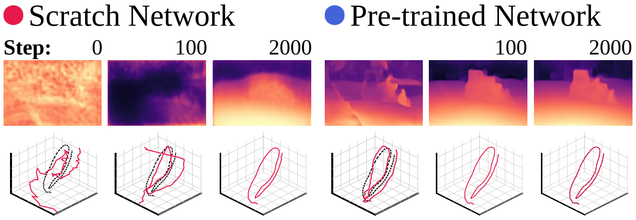


Figure 8. **Effects of pretraining.** While a randomly initialized FlowMap network often provides accurate poses after optimization, pre-training leads to faster convergence and slightly improved poses. Here we plot depth estimates at specific optimization steps (top) as well as pose accuracy with respect to COLMAP during optimization (bottom). Randomly initialized FlowMap networks often require more than 20,000 steps to match the accuracy of a pre-trained initialization at 2,000 steps.

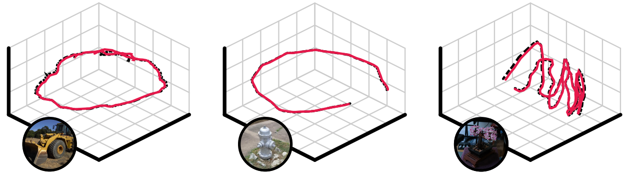


Figure 9. **Qualitative Pose Estimation Comparison.** FlowMap (solid red) recovers camera poses that are very close to those of COLMAP (dotted black).

8. Conclusion.

We have introduced FlowMap, a simple, robust, and scalable first-order method for estimating camera parameters from video. Our model outperforms existing gradient-descent based methods for estimating camera parameters. FlowMap's depth and camera parameters enable subsequent reconstruction via Gaussian Splatting of comparable quality to COLMAP. FlowMap is written in PyTorch and achieves runtimes of 3 minutes for short sequences and 20 minutes for long sequences, and we anticipate that concerted engineering efforts could accelerate FlowMap by an order of magnitude. Perhaps most excitingly, FlowMap is fully differentiable with respect to per-frame depth estimates. FlowMap can thus serve as a building block for a new generation of self-supervised monocular depth estimators, deep-learning-based multi-view-geometry methods, and methods for generalizable novel view synthesis [7, 18, 59, 65, 68, 77], unlocking training on internet-scale datasets of unposed videos.

References

[1] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 7

[2] Wenjing Bian, Zirui Wang, Kejie Li, Jiawang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. 2023. 1, 2, 3

[3] Michael Bloesch, Jan Czarnowski, Ronald Clark, Stefan Leutenegger, and Andrew J Davison. Codeslam—learning a compact, optimisable representation for dense visual slam. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2560–2568, 2018. 2

[4] Richard Strong Bowen, Richard Tucker, Ramin Zabih, and Noah Snavely. Dimensions of motion: Monocular prediction through flow subspaces. In *Proceedings of the International Conference on 3D Vision (3DV)*, pages 454–464. IEEE, 2022. 2

[5] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. In *ECCV*, 2024. 3

[6] Carlos Campos, Richard Elvira, Juan J. Gómez Rodríguez, José M. M. Montiel, and Juan D. Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *Transactions on Robotics*, (6):1874–1890, 2021. 2

[7] David Charatan, Sizhe Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 8

[8] Yu Chen and Gim Hee Lee. Dbarf: Deep bundle-adjusting generalizable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24–34, 2023. 3

[9] Zezhou Cheng, Carlos Esteves, Varun Jampani, Abhishek Kar, Subhransu Maji, and Ameesh Makadia. Lu-nerf: Scene and pose estimation by synchronizing local unposed nerfs. *arXiv preprint arXiv:2306.05410*, 2023. 3

[10] Shin-Fang Chng, Sameera Ramasinghe, Jamie Sherrah, and Simon Lucey. Garf: gaussian activated radiance fields for high fidelity reconstruction and pose estimation. *arXiv e-prints*, pages arXiv–2204, 2022. 3

[11] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proc. CVPR*, 2020. 4, 5

[12] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. *Advances in neural information processing systems*, 29, 2016. 2

[13] Ronald Clark, Michael Bloesch, Jan Czarnowski, Stefan Leutenegger, and Andrew J Davison. Learning to solve nonlinear least squares for monocular stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 284–299, 2018. 2

[14] Jan Czarnowski, Tristan Laidlow, Ronald Clark, and Andrew J. Davison. Deepfactors: Real-time probabilistic dense monocular SLAM. *Computing Research Repository (CoRR)*, 2020. 2

[15] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised NeRF: Fewer views and faster training for free. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[16] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018. 1

[17] Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. *arXiv preprint arXiv:2306.08637*, 2023. 2

[18] Yilun Du, Cameron Smith, Ayush Tewari, and Vincent Sitzmann. Learning to render novel views from wide-baseline stereo pairs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 8

[19] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 834–849. Springer, 2014. 2

[20] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence*, 40(3):611–625, 2017. 2

[21] Hongyu Fu, Xin Yu, Lincheng Li, and Li Zhang. Cbarf: Cascaded bundle-adjusting neural radiance fields from imperfect camera poses, 2023. 3

[22] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A. Efros, and Xiaolong Wang. Colmap-free 3d gaussian splatting. 2023. 3

[23] Yang Fu, Ishan Misra, and Xiaolong Wang. Mononerf: Learning generalizable nerfs from monocular videos without camera poses. 2023. 3

[24] Zelin Gao, Weichen Dai, and Yu Zhang. Adaptive positional encoding for bundle-adjusting neural radiance fields. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3261–3271, 2023. 3

[25] Adam W. Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2

[26] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Anima Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 5846–5854, 2021. 3

[27] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker: It is better to track together. 2023. 2

[28] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. Splatam: Splat, track & map 3d gaussians for dense rgb-d slam. *arXiv preprint arXiv:2312.02126*, 2023. 3

[29] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG)*, 42(4):1–14, 2023. 2, 3, 6

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014. 6

[31] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (TOG)*, 36 (4), 2017. 7

[32] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1611–1621, 2021. 3

[33] Zihang Lai, Sifei Liu, Alexei A Efros, and Xiaolong Wang. Video autoencoder: self-supervised disentanglement of static 3d structure and motion. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 9730–9740, 2021. 3

[34] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[35] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 5741–5751, 2021. 1, 2, 3, 7

[36] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. LightGlue: Local Feature Matching at Light Speed. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023. 1

[37] Chao Liu, Jinwei Gu, Kihwan Kim, Srinivasa G Narasimhan, and Jan Kautz. Neural rgb (r) d sensing: Depth and uncertainty from a video camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10986–10995, 2019. 2

[38] Yu-Lun Liu, Chen Gao, Andreas Meuleman, Hung-Yu Tseng, Ayush Saraf, Changil Kim, Yung-Yu Chuang, Johannes Kopf, and Jia-Bin Huang. Robust dynamic radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13–23, 2023. 3

[39] Zixin Luo, Tianwei Shen, Lei Zhou, Siyu Zhu, Runze Zhang, Yao Yao, Tian Fang, and Long Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 168–183, 2018. 2

[40] Hidenobu Matsuki, Riku Murai, Paul H. J. Kelly, and Andrew J. Davison. Gaussian Splatting SLAM. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[41] Andreas Meuleman, Yu-Lun Liu, Chen Gao, Jia-Bin Huang, Changil Kim, Min H. Kim, and Johannes Kopf. Progressively optimized local radiance fields for robust view synthesis. In *CVPR*, 2023. 3

[42] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 7

[43] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–421, 2020. 3

[44] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017. 2

[45] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (TOG)*, 41(4):102:1–102:15, 2022. 3

[46] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *Transactions on Robotics*, 33(5):1255–1262, 2017. 2

[47] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *Transactions on Robotics*, (5):1147–1163, 2015. 2

[48] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3504–3515, 2020. 3

[49] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: Learning local features from images. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018. 2

[50] Keunhong Park, Philipp Henzler, Ben Mildenhall, Jonathan T Barron, and Ricardo Martin-Brualla. Camp: Camera preconditioning for neural radiance fields. *ACM Transactions on Graphics (TOG)*, 42(6):1–11, 2023. 7

[51] 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 *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 10901–10911, 2021. 7

[52] Antoni Rosinol, Marcus Abate, Yun Chang, and Luca Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1689–1696. IEEE, 2020. 2

[53] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4938–4947, 2020. 1, 2

[54] Paul-Edouard Sarlin, Ajaykumar Unagar, Måns Larsson, Hugo Germain, Carl Toft, Viktor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, and Torsten Sattler. Back to the Feature: Learning Robust Camera Localization from Pixels to Pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

[55] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. 1, 2, 6

[56] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhöfer. Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[57] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 3

[58] Cameron Smith, Yilun Du, Ayush Tewari, and Vincent Sitzmann. Flowcam: Training generalizable 3d radiance fields without camera poses via pixel-aligned scene flow. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 3, 4, 5

[59] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Generalizable patch-based neural rendering. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2022. 8

[60] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. In *ACM Transactions on Graphics (TOG)*, 2023. 7

[61] Chengzhou Tang and Ping Tan. BA-Net: Dense bundle adjustment network. *arXiv preprint arXiv:1806.04807*, 2018. 2

[62] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. *arXiv preprint arXiv:1812.04605*, 2018. 2

[63] Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[64] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 2021. 2

[65] Ayush Tewari, Tianwei Yin, George Cazenavette, Semon Rezchikov, Joshua B Tenenbaum, Frédo Durand, William T Freeman, and Vincent Sitzmann. Diffusion with forward models: Solving stochastic inverse problems without direct supervision. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 8

[66] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5038–5047, 2017. 2

[67] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Visual geometry grounded deep structure from motion. *arXiv preprint arXiv:2312.04563*, 2023. 3

[68] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 8

[69] Wenshan Wang, Yaoyu Hu, and Sebastian Scherer. Tartanvo: A generalizable learning-based vo. In *Conference on Robot Learning*, pages 1761–1772. PMLR, 2021. 2

[70] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. NeRF−−: Neural radiance fields without known camera parameters. *arXiv:2102.07064*, 2021. 1

[71] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 3

[72] Xiuchao Wu, Jiamin Xu, Xin Zhang, Hujun Bao, Qixing Huang, Yujun Shen, James Tompkin, and Weiwei Xu. Scanerf: Scalable bundle-adjusting neural radiance fields for large-scale scene rendering. *ACM Transactions on Graphics (TOG)*, 2023.

[73] Yitong Xia, Hao Tang, Radu Timofte, and Luc Van Gool. Sinerf: Sinusoidal neural radiance fields for joint pose estimation and scene reconstruction. *arXiv preprint arXiv:2210.04553*, 2022. 3

[74] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022. 6

[75] Qingsong Yan, Qiang Wang, Kaiyong Zhao, Jie Chen, Bo Li, Xiaowen Chu, and Fei Deng. Cf-nerf: Camera parameter free neural radiance fields with incremental learning. *arXiv preprint arXiv:2312.08760*, 2023. 3

[76] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021. 3

[77] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 8

[78] Vladimir Yugay, Yue Li, Theo Gevers, and Martin R Oswald. Gaussian-slam: Photo-realistic dense slam with gaussian splatting. *arXiv preprint arXiv:2312.10070*, 2023. 3

[79] Zhoutong Zhang, Forrester Cole, Zhengqi Li, Michael Rubinstein, Noah Snavely, and William T Freeman. Structure and motion from casual videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 20–37. Springer, 2022. 3

[80] Wang Zhao, Shaohui Liu, Hengkai Guo, Wenping Wang, and Yong-Jin Liu. Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 523–542. Springer, 2022. 2

[81] Huizhong Zhou, Benjamin Ummenhofer, and Thomas Brox. Deeptam: Deep tracking and mapping. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 822–838, 2018. 2