VGGSfM: Visual Geometry Grounded Deep Structure From Motion

Jianyuan Wang[1,2] Nikita Karaev [1,2] Christian Rupprecht[1] David Novotny[2]

[1]Visual Geometry Group, University of Oxford [2]Meta AI

https://vggsfm.github.io

Abstract

Structure-from-motion (SfM) is a long-standing problem in the computer vision community, which aims to reconstruct the camera poses and 3D structure of a scene from a set of unconstrained 2D images. Classical frameworks solve this problem in an incremental manner by detecting and matching keypoints, registering images, triangulating 3D points, and conducting bundle adjustment. Recent research efforts have predominantly revolved around harnessing the power of deep learning techniques to enhance specific elements (e.g., keypoint matching), but are still based on the original, non-differentiable pipeline. Instead, we propose a new deep pipeline VGGSfM, where each component is fully differentiable and thus can be trained in an end-to-end manner. To this end, we introduce new mechanisms and simplifications. First, we build on recent advances in deep 2D point tracking to extract reliable pixel-accurate tracks, which eliminates the need for chaining pairwise matches. Furthermore, we recover all cameras simultaneously based on the image and track features instead of gradually registering cameras. Finally, we optimise the cameras and triangulate 3D points via a differentiable bundle adjustment layer. We attain state-of-the-art performance on three popular datasets, CO3D, IMC Phototourism, and ETH3D.

1. Introduction

Reconstructing the camera parameters and the 3D structure of a scene from a set of unconstrained 2D images is a long-standing problem in the computer vision community. Among many other applications [8, 27, 29, 53, 79], it has recently emerged as an important component of learning neural fields [10, 28, 34, 37, 48, 76]. The problem is usually solved via the Structure-from-Motion (SfM) framework which estimates the 3D point cloud (Structure) and the parameters of each camera (Motion) in the scene. State-of-the-art methods [25, 39] follow the incremental SfM paradigm whose origins can be traced back to the early 2000s [23, 58]. It usually begins with a small set of correspondence-rich images as initialization, and gradually adds more views into the reconstruction, through keypoint detection, matching, verification, image registration, triangulation, bundle adjustment (BA), and so on [2, 20, 26, 59].

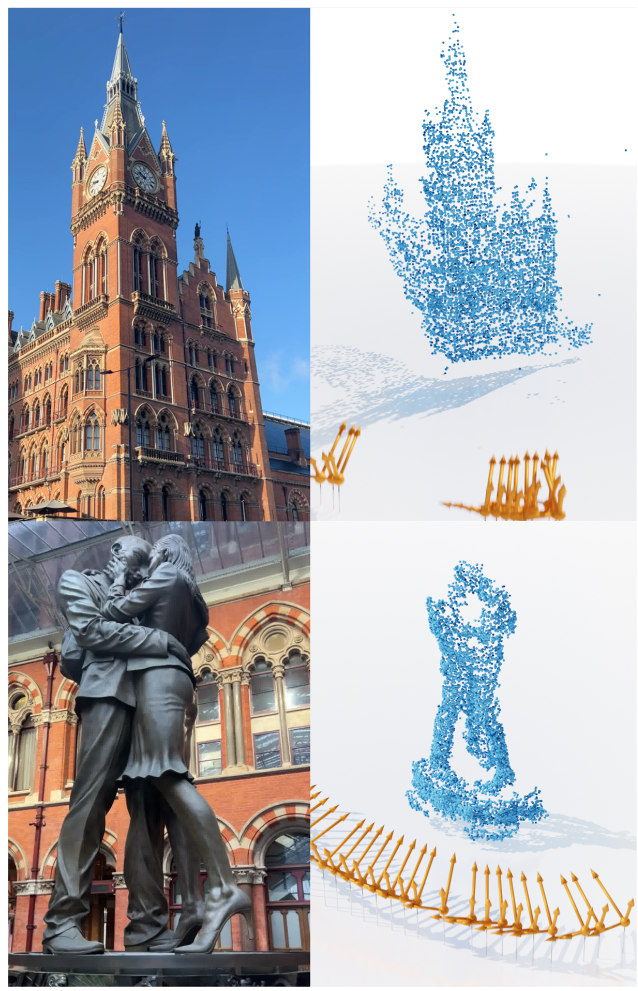

Figure 1. **Reconstruction of In-the-wild Photos with VGGSfM**, displaying estimated point clouds (in blue) and cameras (orange).

Recent research efforts have predominantly revolved around leveraging the power of deep learning techniques to

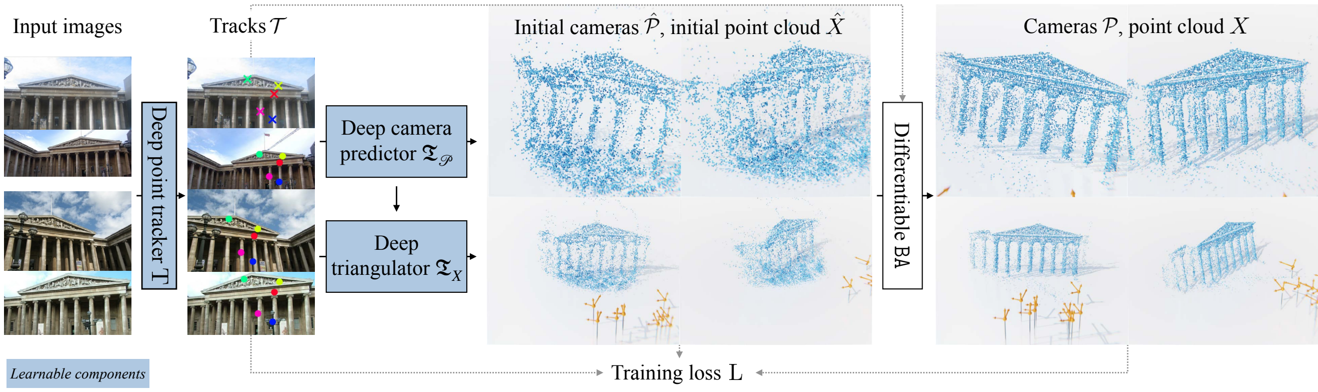

Figure 2. **Overview of VGGSfM.** Our method extracts 2D tracks from input images, reconstructs cameras using image and track features, initializes a point cloud based on these tracks and camera parameters, and applies a bundle adjustment layer for reconstruction refinement. The whole framework is fully differentiable and designed for end-to-end training.

enhance specific elements within the original pipeline while preserving the incremental SfM framework as a whole. For instance, SuperPoint and SuperGlue [14, 57] focus on improving keypoint detection and matching. Pixel-perfect SfM [39] proposes deep feature-metric refinement to adjust both keypoints and bundles. Detector-free feature matching methods [12, 64, 75] bypass early keypoint detection by means of attention, which is powerful in poorly textured scenes. Detector-free SfM [25] builds a coarse SfM model through quantized detector-free matches and then iteratively refines it with multi-view consistency constraints. These advancements successfully combine deep learning approaches (such as deep feature matching) with well-established hand-engineered components, such as the incremental camera registration of COLMAP [59].

The widespread success of end-to-end training warrants the question of what benefits it can bring to long-standing frameworks such as SfM. Naturally, it is often difficult to assess the merits of new approaches when compared with decades of continuous improvements. Nonetheless, in this paper, we answer this question by introducing a fully-differentiable SfM pipeline, dubbed Visual Geometry Grounded Deep Structure From Motion (VGGSfM), which trains in an end-to-end manner. We find that this allows the pipeline to be *simpler* than prior frameworks while achieving better or comparable performance. Training end-to-end allows each component to generate outputs that facilitate the task of its successor.

To build a fully-differentiable pipeline, we make several substantial changes to the SfM procedures and overall obtain better performance. Specifically, our model builds on recent advances in deep 2D point tracking [15, 16, 22, 32] to directly extract reliable pixel-accurate tracks. This simplifies the correspondence estimation step in traditional SfM, which first estimates pairwise matches and then connects them into tracks. Then, based on the image and track features, VGGSfM estimates all cameras jointly via a Transformer [72], and subsequently all 3D points. Different from *Incremental* SfM, this approach is simpler and easier to differentiate as it does not depend on a discrete, combinatorial correspondence chaining step. Finally, for bundle adjustment, we replace the commonly employed non-differentiable Ceres solver [3] with the fully differentiable second-order Theseus solver [54].

Hence, we fuse all the SfM components into a single fully differentiable reconstruction function f. Besides that, in our experiments, we also show that the individual modules perform well in isolation. Ultimately, end-to-end training yields another performance improvement, that surpasses the performance of isolated components.

We evaluate VGGSfM for the task of camera pose estimation on the CO3Dv2 [55] and IMC Phototourism [31] datasets, and for 3D triangulation on the ETH3D [60] dataset. Our method attains strong performance on all benchmarks. At the same time, we conduct in-the-wild reconstruction to validate the generalization ability of our proposed framework, as shown in Fig. 1.

2. Related Work

Structure from Motion is a fundamental problem in computer vision and has been investigated for decades [23, 52, 53]. The classical pipelines usually solve the SfM problem in a global [13, 50, 80] or incremental [2, 19, 59, 63, 81] manner. Both of which are usually based on pairwise image keypoint matching. Incremental SfM is arguably the most widely adopted strategy (*e.g.*, the popular framework COLMAP [59]). Therefore, in the following sections, we refer to incremental SfM as "classical" or "traditional" SfM. We defer the discussion of global SfM to the supplementary.

Traditional SfM frameworks often start by detecting keypoints and feature descriptors [4, 42, 43, 46]. They then search for image pairs with overlapping frusta by matching these keypoints across different images (*e.g.*, with a nearest-neighbour search) [2, 41, 59]. These image pairs are further verified via two-view epipolar geometry or homography [23] through RANSAC [18]. Then, a pair or a small set of images is carefully selected for initialization. New images are gradually registered by solving the Perspective-n-Point (PnP) problem [44], followed by triangulating 3D points, and bundle adjustment [69]. This process is iterated until all the frames are either registered or discarded. 2D correspondences (multi-view tracks) are the basis of the whole process, however, they are usually simply constructed by chaining two-view matches [59].

Many deep-learning approaches have been proposed to enhance this framework. For example, [14, 70, 84] provide better keypoint detection and [11, 30, 40, 57, 61] focus on matching. Furthermore, detector-free matching methods [12, 64, 75] propose to avoid sparse keypoint detection by building semi-dense matches via self and cross attention. [7] solves the subproblems of SfM by graph attention networks. Some studies improve the performance of RANSAC by making it trainable [5, 6, 77]. Recent state-of-the-art methods are PixSfM [39] and the concurrent Detector-free SfM (DFSfM) [25]. PixSfM refines the tracks and structure estimated by COLMAP through feature-metric keypoint adjustment and feature-metric bundle adjustment. Detector-free SfM proposes to first build a coarse SfM model using detector-free matches and COLMAP (or other frameworks), and then to iteratively refine the tracks and the structure of the coarse model by enforcing multi-view consistency.

Recently, fully differentiable SfM pipelines have also been explored. They usually use deep neural networks to regress camera poses and depths [65, 66, 68, 71, 73, 78, 87]. Although using an approximation of bundle adjustment [65, 68, 78], these methods suffer from limited generalizability and scalability (very few input frames) [65, 71, 73, 78, 87], or rely on temporal relationship [66, 68]. Meanwhile, some methods are category-specific [45, 82, 83]. The recent efforts on deep camera pose estimation can scale up to more than 50 frames, but they do not reconstruct the scene [36, 62, 74, 85].

Point tracking. Since VGGSfM proposes a novel point track predictor, next, we review recent advances in this field. Inspired by the optical-flow architecture of RAFT [67], PIPs [22] revisited point tracking, a task related to Particle Video [56], and proposed a highly accurate tracker of isolated points in a video. TAP-Vid [15] (*i.e.*, *"Tracking Any Point"*) introduced a benchmark for point tracking and a baseline model, which was later improved in TAPIR [16] by integrating the iterative update mechanism from PIPs. PointOdyssey [86] simplified PIPs and proposed a benchmark for the long-term version of point tracking. CoTracker [32] closed the gap between single point tracking and dense Optical Flow with joint point tracking. However, these works are designed for videos, *i.e.* temporally-ordered sequences of frames. In our point tracker, given the input frames are unordered, we do not assume a temporal relationship between input frames. We therefore process all frames jointly, avoiding windowed inference of [32]. Since SfM relies on highly accurate correspondences, our tracks are further refined in a coarse-to-fine manner to achieve subpixel accuracy.

3. Method

In this section, we describe the components of VGGSfM and how they are composed in a fully differentiable pipeline. An overview of our framework is shown in Fig. 2.

Problem setting Given a set of free-form images observing a scene, VGGSfM estimates their corresponding camera parameters and the 3D scene shape represented as a point cloud. Formally, given a tuple $\mathcal{I} = (I_1, ..., I_{N_I})$ of $N_I \in \mathbb{N}$ RGB images $I_i \in \mathbb{R}^{3\times H\times W}$, VGGSfM estimates the corresponding camera projection matrices $\mathcal{P} = (P_1, ..., P_{N_I} | P_i \subset \mathbb{R}^{3\times 4})$ and the scene cloud $X = \{\mathbf{x}^j\}_{j=1}^{N_\mathbf{x}}$ of $N_\mathbf{x} \in \mathbb{N}$ 3D points $\mathbf{x}^j \in \mathbb{R}^3$. Each projection matrix P_i consists of extrinsics (pose) $g_i \in \mathbb{SE}(3)$ and intrinsics $K_i \in \mathbb{R}^{3\times 3}$.

A 3D point $\mathbf{x}^j$ can be projected to the i-th camera yielding a 2D screen coordinate $\mathbf{y}_i^j = P_i(\mathbf{x}^j) \sim \lambda K_i \hat{\mathbf{x}}_i^j; \lambda \in \mathbb{R}_+$, where $\hat{\mathbf{x}}_i^j = g_i \mathbf{x}^j$ is the world coordinate $\mathbf{x}^j$ expressed in view-coordinates of the i-th camera. The projection of the point $\mathbf{x}^j$ to all input cameras is a *track* $T^j = \left((y_1^j, v_1^j) ..., (y_{N_I}^j, v_{N_I}^j)\right)$ consisting of N_I matching 2D points $\mathbf{y}_i^j \in \mathbb{R}^2$, and their corresponding binary indicators $v_i^j \in \{0,1\}$ denoting visibility of the j-th point in the i-th camera. We denote $\mathcal{T}_i = \{T_i^1, ..., T_i^{N_T}\}$ as the set of all tracks T_i^j in the i-th camera.

3.1. Method overview

VGGSfM implements SfM via a single function f_θ

$$f_\theta(\mathcal{I}) = \mathcal{P}, X \qquad (1)$$

accepting the set of N_I scene images $\mathcal{I}$ and outputting the camera parameters $\mathcal{P}$ and the scene point cloud X. Importantly, f_θ is fully differentiable and, as such, its parameters θ are learned by minimizing the training loss $\mathcal{L}$:

$$\theta^\star = \arg\min_\theta \sum_{s=1}^{S} \mathcal{L}(f_\theta(\mathcal{I}_s), \mathcal{P}_s^\star, \mathcal{T}_s^\star, X_s^\star), \qquad (2)$$

summing over $S \in \mathbb{N}$ training image sets $\mathcal{I}_s$ annotated with ground-truth cameras $\mathcal{P}_s^\star$, tracks $\mathcal{T}_s^\star$, and point clouds $X_s^\star$. We defer the details of $\mathcal{L}$ to Sec. 3.5 and, in the following paragraphs, discuss the architecture of f_θ.

The reconstruction function Following traditional SfM [59], VGGSfM decomposes the reconstruction function f_θ into four seamless stages: 1) point tracking $\mathtt{T}$, 2) initial camera estimator $\mathfrak{T}_\mathcal{P}$, 3) triangulator $\mathfrak{T}_X$ and, 4) Bundle Adjustment $\mathtt{BA}$, as follows:

$$\begin{aligned}\mathcal{T} &= \mathtt{T}(\mathcal{I}) \\ \hat{\mathcal{P}} &= \mathfrak{T}_\mathcal{P}(\mathcal{I}, \mathcal{T}) \\ \hat{X} &= \mathfrak{T}_X(\mathcal{T}, \hat{\mathcal{P}}) \\ \mathcal{P}, X &= \mathtt{BA}(\mathcal{T}, \hat{\mathcal{P}}, \hat{X}).\end{aligned} \tag{3}$$

The track predictor $\mathtt{T}$, hereafter referred to as "tracker", estimates 2D tracks $\mathcal{T}$ given input images $\mathcal{I}$. Subsequently, $\mathfrak{T}_\mathcal{P}$ and $\mathfrak{T}_X$ provide initial cameras $\hat{\mathcal{P}}$ and an initial point cloud $\hat{X}$ respectively. Finally, $\mathtt{BA}$ enhances accuracy by refining the cameras and 3D points together.

3.2. Tracking

Establishing precise 2D correspondences is important for achieving accurate 3D reconstruction. Traditionally, SfM frameworks first estimate pairwise image-to-image correspondences that are later chained into multi-image tracks T [25, 39, 59]. Here, typically only point-pair matching benefits from learned components [14, 40, 57, 70], while the chaining of pairwise correspondences remains a hand-engineered process.

Instead, VGGSfM *significantly simplifies SfM correspondence tracking* by employing a deep feed-forward tracking function. It accepts a collection of images and directly outputs a set of reliable point trajectories across all images. We achieve this by exploiting recent advances in video point tracking methods [15, 16, 22, 32]. Although developed for video-point tracking, these methods are inherently appropriate for SfM which requires a very compact set of highly accurate tracks (*e.g.*, dense optical flow is too memory-demanding). Furthermore, point trackers mitigate the potential errors (sometimes called drift) caused by chaining of pairwise matches. However, as we describe below, our design differs from video trackers because SfM, which accepts free-form images, cannot assume temporal smoothness or ordering, and requires sub-pixel accuracy.

Tracker architecture The design of our tracker $\mathtt{T}$ follows [22, 32], and is illustrated in Fig. 3. More specifically, given N_T query points $\{\hat{\mathbf{y}}_i^1, ...\hat{\mathbf{y}}_i^{N_T}\}$ in a frame I_i, we bilinearly sample their corresponding query descriptors $\{\mathbf{m}_i^1, ..., \mathbf{m}_i^{N_T}\}$ from image feature maps output by a 2D CNN. Then, each query descriptor is correlated with the feature maps of all N_I frames at different spatial resolutions, which constructs a cost-volume pyramid. Flattening the latter yields tokens $V \in \mathbb{R}^{N_T \times N_I \times C}$, where C is the total number of elements in the cost-volume pyramid. Feeding the tokens to a Transformer, we obtain tracks $\mathcal{T} = \{T^j\}_{j=1}^{N_T}$. Recall that each track T^j comprises the

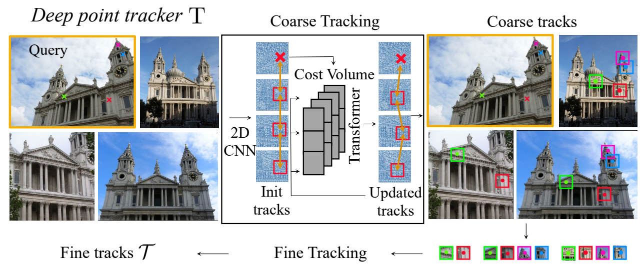

Figure 3. **Architecture of Tracker T.** We adopt a coarse-to-fine design for the tracker. The coarse tracker first locates the approximate positions of corresponding points, and the fine tracker then refines these initial predictions.

set of N_I tracked 2D locations $\mathbf{y}_i^j$ together with predicted visibility indicators v_i^j.

It is worth noting that, differently from [22, 32], our tracker does not assume temporal continuity. Therefore, we avoid the sliding window approach and, instead, attend to all the input frames together. Furthermore, unlike in [32], we predict each track independently of others. This allows to track a larger number of points at test time leading to increased density of reconstructed point clouds.

Predicting tracking confidence In SfM, it is crucial to filter out any outlier correspondences as they can negatively impact the subsequent reconstruction stages. To this end, we enhance the tracker with the ability to estimate confidence for each track-point prediction.

More specifically, we leverage the aleatoric uncertainty [33, 51] model which predicts variance σ_i^j together with each 2D track point $\mathbf{y}_i^j$, so that the resulting normal distribution $\mathcal{N}(\mathbf{y}_i^{j\star}|\mathbf{y}_i^j, \sigma_i^j)$ tightly peaks around each ground-truth 2D track point $\mathbf{y}_i^{j\star}$. Hence, during training, the ℓ_1/ℓ_2 loss, originally used in video point tracking, is replaced with the (negated) logarithm of the latter probability evaluated at each ground truth point $\mathbf{y}_i^{j\star}$. Once trained, the confidence measure $1/\sigma_i^j$ is proportional to the inverse of the predicted variance. In practice, we assume a diagonal covariance matrix resulting in horizontal and vertical uncertainties σ_i^j.

Coarse-to-fine tracking Moreover, since SfM requires highly accurate (pixel or sub-pixel level) correspondences, we employ a coarse-to-fine point-tracking strategy. As described above, we first coarsely track image points using feature maps that fully cover the input images I. Then, we form $P \times P$ patches by cropping input images around the coarse point estimates and execute the tracking again to obtain a sub-pixel estimate. Recall that, differently from the chained matching of traditional SfM, our tracker is fully differentiable. This enables back-propagating the gradient of the training loss $\mathcal{L}$ through the whole framework to the tracker parameters. This reinforces the synergy between the tracking and the ensuing stages, which are described next.

3.3. Learnable camera & point initialization

As discussed above, a traditional SfM pipeline [39, 59] usually relies on an an incremental loop, which often initializes with a correspondence-rich image pair, gradually registers new frames, enlarges the point cloud, and conducts joint optimization (*e.g*., BA). However, although the framework has been fortified in robustness and accuracy through decades of improvements, this cumulative process has inevitably led to increased complexity. Furthermore, Incremental SfM is largely non-differentiable which complicates end-to-end learning from annotated data.

Thus, in pursuit of simplicity and differentiability, our method departs from the classical SfM scheme. Inspired by recent advances in deep camera pose estimators [36, 74, 85], we propose to initialize the cameras and the point cloud with a pair of deep Transformer [72] networks. Importantly, we register all cameras and reconstruct all scene points collectively in a non-incremental differentiable fashion.

Learnable camera initializer The predictor of initial cameras $\hat{\mathcal{P}}$ is implemented as a deep Transformer architecture $\mathfrak{T}_{\mathcal{P}}$:

$$\hat{\mathcal{P}} = \mathfrak{T}_{\mathcal{P}}(\{\phi(I_i)|I_i \in \mathcal{I}\}, \{d^{\mathcal{P}}(y_i^j)|\forall T_i \in \mathcal{T}\ \forall y_i^j \in T_i\}). \tag{4}$$

It accepts a set of tokens comprising global ResNet50 [24] features $\phi(I_i)$ of input images $\mathcal{I}$, and the set of descriptors $d^{\mathcal{P}}(\mathbf{y}_i^j)$ of track points $\mathbf{y}_i^j \in T_i \in \mathcal{T}$. Here, each track descriptor is output by an auxiliary branch of the tracker $\texttt{T}$. Given these inputs, $\mathfrak{T}_{\mathcal{P}}$ first applies cross-attention between the global image feature (query) and the track-descriptor (key-value) pairs yielding $N_I = |\mathcal{I}|$ tokens per scene. The output of cross-attention is then concatenated with an embedding of a preliminary camera estimated by the 8-point algorithm taking the correspondences between track points $\mathbf{y}_i^j$ as input. Finally, we feed this concatenation to a Transformer trunk resulting in the initial cameras $\hat{\mathcal{P}}$.

Learnable triangulation Given initial cameras $\hat{\mathcal{P}}$ and 2D tracks $\mathcal{T}$, the triangulator outputs the initial point cloud $\hat{X}$. Similar to the camera predictor, the triangulator is a Transformer $\mathfrak{T}_X$

$$\hat{X} = \mathfrak{T}_X(\{d^X(y_i^j)|\forall T_i \in \mathcal{T} \forall y_i^j \in T_i\}) \tag{5}$$

accepting descriptors $d^X(\mathbf{y}_i^j)$ each comprising a tracker feature, and a positional harmonic embedding [47] of points $\bar{\mathbf{x}}^j \in \bar{X}$ from a preliminary point cloud $\bar{X}$. The preliminary point cloud is formed via closed-form multi-view Direct Linear Transform (DLT) 3D triangulation [23] of the tracks $\mathcal{T}$ given the initial cameras $\hat{\mathcal{P}}$. Please refer to the supplementary for a detailed description of both initializers.

3.4. Bundle adjustment

Given the tracks $\mathcal{T}$ (Sec. 3.2), initial cameras $\hat{\mathcal{P}}$, and the initial point cloud $\hat{X}$ (Sec. 3.3), Bundle Adjustment $\texttt{BA}$ minimizes the reprojection loss $\mathcal{L}_{\text{BA}}$[1, 2, 23, 59]:

$$X, \mathcal{P} = \texttt{BA}(\mathcal{T}, \hat{X}, \hat{\mathcal{P}}) = \arg\min_{X,\mathcal{P}} \mathcal{L}_{\text{BA}}$$
$$\mathcal{L}_{\text{BA}} = \sum_{i=1}^{N_I} \sum_{j=1}^{N_{\mathbf{x}}} v_i^j \|P_i(\mathbf{x}^j) - y_i^j\|, \tag{6}$$

summing over all reprojection errors $\|P_i(\mathbf{x}^j) - \mathbf{y}_i^j\|$ each comprising the distance between the projection $P_i(\mathbf{x}^j)$ of the point cloud $\mathbf{x}^j \in X$ to camera $P_i \in \mathcal{P}$, and the i-th 2D point $\mathbf{y}_i^j \in T^j$ of the track T^j. Additionally, the error terms are filtered out if the corresponding points have low visibility, low confidence, or do not fit the geometric constraints defined by [59]. Points with large reprojections errors are also filtered [59, 63, 81]. More details are provided in the supplementary material.

Differentiable Levenberg-Marquardt Following common practice [39, 59], we minimize Eq. (6) with second-order Levenberg-Marquardt (LM) optimizer [49]. However, optimizing the main training loss (Eq. (2)) via backpropagation requires differentiability of Eq. (6) which is non-trivial. Therefore, we leverage the recently proposed Theseus library [54] which exploits the implicit function theorem to backpropagate through deep networks with nested optimization loops.

3.5. Method details

Camera parameterization Each camera pose $P \in \mathcal{P}$ is parameterized with 8-degrees of freedom: the quaternion $q(R) \in \mathbb{R}^4$ of the rotation $R \in \mathbb{SO}(3)$ and the translation $\mathbf{t} \in \mathbb{R}^3$ components of P's extrinsics $g \in \mathbb{SE}(3)$, and the logarithm $\ln(\mathfrak{f}) \in \mathbb{R}$ of the camera focal length $\mathfrak{f} \in \mathbb{R}^+$. Given these values, the 3×4 pose matrix is defined as $P = KR[\mathbb{I}_3|\mathbf{t}]$, with the calibration matrix $K = [\mathfrak{f}, 0, p_x; 0, \mathfrak{f}, p_y; 0, 0, 1] \subset \mathbb{R}^{3\times3}$ (row major order). Following standard practice [39, 59], we set the principal-point coordinates $p_x, p_y \in \mathbb{R}$ to the center of the image.

Training loss The training loss $\mathcal{L}$ (Eq. (2)) is defined as:

$$\mathcal{L}(f_\theta(\mathcal{I}), \mathcal{P}^\star, \mathcal{T}^\star, X^\star) = \sum_{j=1}^{N_T} |\mathbf{x}^{\star j} - \mathbf{x}^j|_\epsilon + |\mathbf{x}^{\star j} - \hat{\mathbf{x}}^j|_\epsilon +$$
$$+ \sum_{i=1}^{N_I} e_{\mathcal{P}}(P_i^\star, P_i) + e_{\mathcal{P}}(P_i^\star, \hat{P}_i) - \sum_{i=1}^{N_I}\sum_{j=1}^{N_T} \log \mathcal{N}(\mathbf{y}_i^{j\star}|\mathbf{y}_i^j, \sigma_i^j) \tag{7}$$

Here, $|\mathbf{x}^{\star j} - \mathbf{x}^j|_\epsilon$ and $|\mathbf{x}^{\star j} - \hat{\mathbf{x}}^j|_\epsilon$ evaluate the ϵ-thresholded pseudo-Huber loss $|\cdot|_\epsilon$ [9] between the ground truth 3D points $\mathbf{x}^{\star j}$ and the initial and BA-refined 3D points $\hat{\mathbf{x}}^j \in X$, $\mathbf{x}^j \in \hat{X}$ respectively. The camera errors $e_{\mathcal{P}}(P_i^\star, P_i)$ and $e_{\mathcal{P}}(P_i^\star, \hat{P}_i)$ compare the predicted initial pose $\hat{P}_i \in \hat{\mathcal{P}}$ and bundle-adjusted camera pose $P_i \in \mathcal{P}$ to the ground truth camera annotation $P_i^\star \in \mathcal{P}_i^\star$. Here, $e_{\mathcal{P}}(P, P')$ is defined as

Figure 4. **Camera and point-cloud reconstructions** of VGGSfM on Co3D (left) and IMC Phototourism (right).

the Huber-loss $|\cdot|_\epsilon$ between the parameterizations of poses P, P'. Finally, $\log \mathcal{N}(\mathbf{y}_i^{j\star}|\mathbf{y}_i^j, \sigma_i^j)$ computes the likelihood of a ground-truth track point $\mathbf{y}_i^{j\star} \in T_i^\star$ under a probabilistic track-point estimate defined by a 2D gaussian with mean and variance predictions $\mathbf{y}_i^j$ and σ_i^j respectively (i.e. the aleatoric uncertainty model described in Sec. 3.2).

4. Experiments

In this section, we first introduce the datasets together with the protocols for training and evaluation. Then, we provide comparison to existing methods and ablation studies.

Datasets. Following prior work [25, 39, 74], we evaluate camera pose estimation on Co3Dv2 [55] and IMC Phototourism datasets [31], and 3D triangulation on ETH3D [60]. Co3D is an object-centric dataset comprising turntable-style videos from 51 categories of MS COCO [38]. IMC Phototourism, provided by Image Matching Challenge [31], contains 8 testing scenes and 3 validation scenes of famous landmarks. Generally, the Co3D scenes have much wider baselines, making them challenging for traditional frameworks such as COLMAP, while the IMC samples often have sufficiently overlapping fursta, which is where COLMAP excels. ETH3D provides highly accurate point clouds (captured by laser scanner) for 13 indoor and outdoor scenes, and hence is suitable for the evaluation of triangulation.

Training. For the model evaluated on IMC Phototourism and ETH3D, we follow the protocol of [25, 39, 57] and train on the MegaDepth dataset [35]. MegaDepth contains 1M crowd-sourced images depicting 196 tourism landmarks, auto-annotated with SfM tools. Hyper-parameters are tuned using the IMC validation set. As in prior work [25, 40], some scenes of MegaDepth are excluded from training due to their low quality or due to overlap with the IMC test set. For Co3Dv2, we conduct training and evaluation on 41 categories as in [36, 74, 85].

We chose a multi-stage training strategy for better stability. We first train the tracker $\mathtt{T}$ on the synthetic Kubric dataset [21] following the training protocol of [15, 32]. Then, the tracker is fine-tuned solely on Co3D or MegaDepth, depending on the test dataset. Subsequently, we train the camera initializer, with the tracker frozen. Next, the triangulator is trained with the aforementioned two components held frozen. Finally, all components are trained end-to-end. In all stages, we randomly sample a training batch of 3 to 30 frames.

Testing. Given input test frames $\mathcal{I}$, we first select the query frame by identifying the image that is closest to all others based on the cosine similarity between global descriptors extracted by an off-the-shelf ResNet50 [24]. Then, we extract SuperPoint and SIFT keypoints from the query frame to serve as the query points for the tracker $\mathtt{T}$. Although our method can track any query point, it performs better when the queries are distinctive. To improve accuracy, we iterate the whole reconstruction function f_θ multiple times until reaching sub-pixel BA reprojection error $\mathcal{L}_{\mathtt{BA}}$. After the first iteration, the query image for each subsequent iteration is the one that is farthest from the

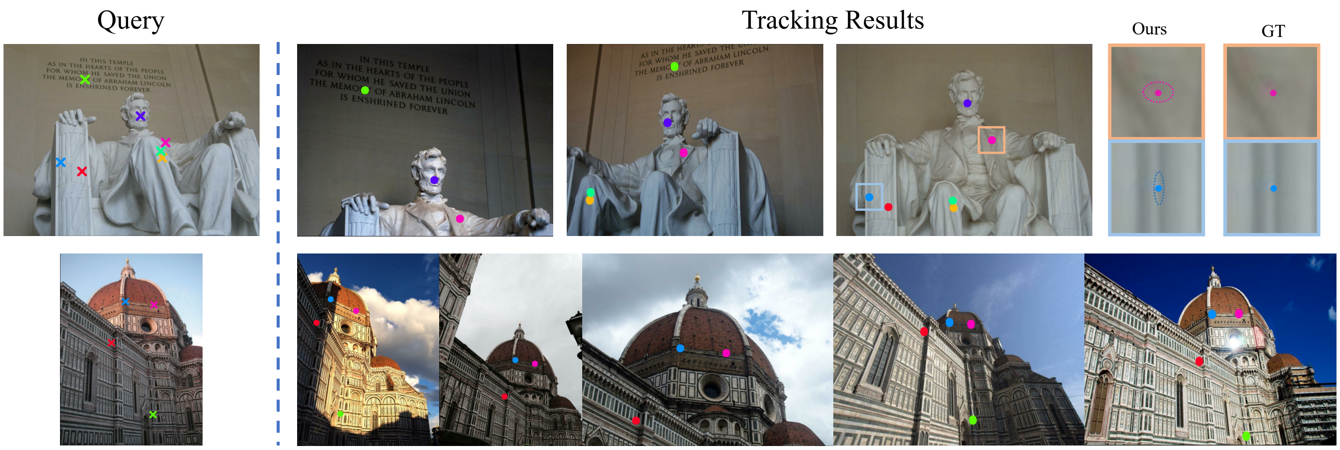

Figure 5. **Qualitative Evaluation of Tracking.** In each row, the left-most frame contains the query image with query points (crosses). The predicted track points $\mathbf{y}_i^j$ (dots) are shown to the right. The top-right part also highlights our track-point confidence predictions (described in Sec. 3.2), illustrated as ellipses with extent proportional to the predicted variance σ_i^j. Note how the uncertainty corresponds to an expectation, *e.g.*, a keypoint covering a vertical stripe has higher uncertainty along the y-axis.

Co3D Dataset	Incremental SfM		Deep					
Method	COLMAP (SP + SG)	PixSfM (SP + SG)	RelPose[85]	PoseReg[74]	RelPose++[36]	PoseDiffusion[74]	Ours w/o Joint	Ours
RRE@15°	31.6	33.7	57.1	53.2	82.3	80.5	<u>88.2</u>	**92.1**
RTE@15°	27.3	32.9	-	49.1	77.2	79.8	<u>83.4</u>	**88.3**
AUC@30°	25.3	30.1	-	45.0	65.1	66.5	<u>70.7</u>	**74.0**

Table 1. **Camera Pose Estimation on Co3D,** where the proposed method outperforms previous methods by a large margin. Ours w/o Joint indicates the variant of our framework without training all components jointly.

IMC Dataset	Method	AUC@3°	AUC@5°	AUC@10°
Deep	DeepSFM	10.27	19.36	31.35
	PoseDiffusion	12.31	23.17	36.82
Incremental SfM	COLMAP (SIFT+NN)	23.58	32.66	44.79
	PixSfM (SIFT + NN)	25.54	34.80	46.73
	PixSfM (LoFTR)	44.06	56.16	69.61
	PixSfM (SP + SG)	45.19	57.22	70.47
	DFSfM (LoFTR)	**46.55**	<u>58.74</u>	<u>72.19</u>
Deep	Ours w/o Joint	38.23	51.60	68.35
	Ours	<u>45.23</u>	**58.89**	**73.92**

Table 2. **Camera Pose Estimation on IMC.** Our method achieves better accuracy than state-of-the-art Incremental SfM approaches on 2 out of 3 AUC thresholds.

query image of the previous iteration (as measured by the ResNet50 cosine similarity). The re-projections of the point cloud from the current iteration initialize the tracking in the next iteration.

4.1. Camera pose estimation

Following [25, 31, 39, 74], we report the metric area-under-curve (AUC) to evaluate camera pose accuracy. In Co3D, similar to [74], we also report the relative rotation error (RRE) and relative translation error (RTE). More specifically, for every pair of input frames, we compute the angular translation and rotation error, which are later compared to a threshold yielding accuracies RTE and RRE respectively. For a range of thresholds, AUC picks the lower between RRE and RTE, and outputs the area under the accuracy-threshold curve. The results on IMC and CO3D are presented in Tab. 1 and Tab. 2 respectively. For a fair comparison on IMC, we finetune DeepSFM [78] and PoseDiffusion [74] on MegaDepth using their open-source code. The results of Incremental SfM methods are copied from Detector-free SfM [25].

Results indicate that VGGSfM outperforms existing methods by a large margin (+9 accuracy points for each metric) on the CO3D dataset. Here, traditional SfM pipelines suffer because of the wide baselines between test frames. On IMC, with a good overlap between views, traditional SfM remains superior to recent data-driven deep-learning pipelines [74, 78]. Our end-to-end trained VGGSfM, however, outperforms all other methods on AUC@10 and AUC@5, and ranks second on AUC@3, convincingly demonstrating its ability to perform well in both narrow- and wide-baseline regimes. The accuracy and completeness of our point clouds can be further qualitatively evaluated in Fig. 4.

Method	Accuracy (%)			Completeness (%)		
	1cm	2cm	5cm	1cm	2cm	5cm
PatchFlow (LoFTR)	66.73	78.73	89.93	3.48	11.34	30.96
PixSfM (LoFTR)	74.42	84.08	92.63	2.91	9.39	27.31
PixSfM (SIFT + NN)	76.18	85.60	93.16	0.17	0.71	3.29
PixSfM (SP + SG)	79.01	87.04	93.80	0.75	2.77	11.28
DFSfM (LoFTR)	80.38	89.01	95.83	3.73	11.07	29.54
Ours	**80.62**	**89.49**	**96.52**	**4.52**	**13.11**	**33.96**

Table 3. **3D Triangulation on ETH3D [60]** reporting the accuracy and completeness metrics at different thresholds.

4.2. 3D triangulation

We evaluate the accuracy and completeness of 3D triangulation on the ETH3D dataset [60] using the same protocol as [17, 25, 39], which triangulates points with fixed camera poses and intrinsics. Results are shown in Tab. 3, where metrics are averaged over all scenes. Our VGGSfM achieves better accuracy and completeness than all baselines (PatchFlow [17], PixSfM [39], and DFSfM [25]), regardless of which keypoint detection or matching method they use. This is especially obvious from the completeness attained at the 5cm threshold, with our 33.96% compared to 29.54% of the best prior work.

4.3. Ablation study

End-to-end Training. As reported in Tab. 1 and Tab. 2, the end-to-end joint training of the whole framework is important for achieving state-of-the-art performance. Specifically, comparing VGGSfM to an ablation which lacks end-to-end training (Ours w/o joint) we record an improvement from 70.7% AUC@30 to 74.0% on the Co3D dataset, and 68.35% AUC@10 to 73.92% on the IMC dataset. This demonstrates the benefits of our fully-differentiable design, and the synergy between its components.

Tracking or Pairwise Matching. We compare the performance of our predicted tracks to pairwise matching methods on the IMC dataset. Specifically, we split our 2D tracks into pairwise matches and feed these matches to PixSfM. Also, we construct 2D tracks by pairwise matching (based on the open-source implementation of PixSfM) and feed them to our framework. It is worth noting that tracks from pairwise matching have a lot of "holes" because pairwise matching cannot guarantee proper point tracking. We fix these holes by setting their locations as the point locations in the query frame, marking them as invisible. At the same time, for fair comparison, cameras are still initialized using our tracks, because SP+SG cannot provide track features to our camera initializer. The results are shown in Tab. 4. Although COLMAP (the basis of PixSfM) is designed and carefully engineered for pairwise matching, our tracks achieve a slightly better result than the state-of-the-art matching option SP+SG. Instead, directly feeding SP+SG tracks to our framework leads to a performance drop. We attribute this to the fact that using SP+SG tracks cannot benefit from the joint training. We also provide a qualitative evaluation of our tracking accuracy in Fig. 5 on the IMC dataset.

	PixSfM(SP + SG)	PixSfM(Our Tracks)	Ours(SP + LG)	Ours
AUC@10°	70.47	70.62	68.78	73.92

Table 4. **Tracking or Pairwise Matching.** We respectively provide tracks predicted by our tracker or matches estimated by SP+SG to PixSfM and to our VGGSfM.

	PoseDiffusion	Our Camera Initializer
DLT	62.18	69.42
Our Triangulator	66.37	73.92

Table 5. **Ablation Study for Camera Initializer and Triangulator.** A clear performance drop is observed when replacing our camera initializer by deep camera prediction method PoseDiffusion, or replacing triangulator by DLT.

Camera Initializer and Triangulator. We also validate the design of our camera initializer $\mathfrak{T}_{\mathcal{P}}$ and triangulator $\mathfrak{T}_X$ on the IMC dataset. As reported in Tab. 5, AUC@10 drops clearly if we replace them with alternatives, proving that they provide sufficiently accurate initialization for our global bundle adjustment BA, without the need for incremental camera registration.

Coarse-to-fine Tracking. As dicussed above, accurate correspondences are important for structure from motion. We demonstrate the significance of our coarse-to-fine tracking mechanism for the method performance. By conducting an ablation study where the fine tracker is removed, we observe a significant performance drop on the IMC dataset, with AUC@10 dropping from 73.92% to 62.30%.

5. Conclusion

In this paper, we have presented VGGSfM, a fully differentiable SfM approach. We find that even long-standing pipelines, such as Structure-from-Motion, benefit from a learned adaptation between their components. This allows VGGSfM to be simpler than traditional SfM frameworks while achieving better performance across benchmark datasets. Moreover, our framework is fully implemented in Python, which will allow for easy modification and improvements in the future. While VGGSfM already achieves good performance, it cannot yet compete with established pipelines in all application domains. For example, it currently lacks the capability to process thousands of images as in traditional SfM frameworks. Nonetheless, we find differentiable SfM a promising direction of research, and our approach lays the foundation for further advances.

References

[1] Sameer Agarwal, Noah Snavely, Steven M Seitz, and Richard Szeliski. Bundle adjustment in the large. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part II 11*, pages 29–42. Springer, 2010. 5

[2] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54 (10):105–112, 2011. 1, 2, 3, 5

[3] Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres Solver, 2022. 2

[4] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-Up Robust Features (SURF). *CVIU*, 110(3), 2008. 2

[5] Eric Brachmann and Carsten Rother. Neural-guided ransac: Learning where to sample model hypotheses. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4322–4331, 2019. 3

[6] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6684–6692, 2017. 3

[7] Lucas Brynte, José Pedro Iglesias, Carl Olsson, and Fredrik Kahl. Learning structure-from-motion with graph attention networks. *arXiv preprint arXiv:2308.15984*, 2023. 3

[8] Jonathan L Carrivick, Mark W Smith, and Duncan J Quincey. *Structure from Motion in the Geosciences*. John Wiley & Sons, 2016. 1

[9] Pierre Charbonnier, Laure Blanc-féraud, Gilles Aubert, and Michel Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Trans. Image Processing*, 6:298–311, 1997. 5

[10] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*, pages 333–350. Springer, 2022. 1

[11] Hongkai Chen, Zixin Luo, Jiahui Zhang, Lei Zhou, Xuyang Bai, Zeyu Hu, Chiew-Lan Tai, and Long Quan. Learning to match features with seeded graph matching network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6301–6310, 2021. 3

[12] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David Mckinnon, Yanghai Tsin, and Long Quan. Aspanformer: Detector-free image matching with adaptive span transformer. In *European Conference on Computer Vision*, pages 20–36. Springer, 2022. 2, 3

[13] Zhaopeng Cui and Ping Tan. Global structure-from-motion by similarity averaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 864–872, 2015. 2

[14] 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. 2, 3, 4

[15] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens Continente, Kucas Smaira, Yusuf Aytar, Joao Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. In *NeurIPS Datasets Track*, 2022. 2, 3, 4, 6

[16] 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, 3, 4

[17] Mihai Dusmanu, Johannes L Schönberger, and Marc Pollefeys. Multi-view optimization of local feature geometry. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 670–686. Springer, 2020. 8

[18] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 3

[19] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pages 368–381. Springer, 2010. 2

[20] Yasutaka Furukawa, Brian Curless, Steven M. Seitz, and Richard Szeliski. Towards Internet-scale multi-view stereo. In *Proc. CVPR*. IEEE, 2010. 1

[21] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: a scalable dataset generator. 2022. 6

[22] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *ECCV*, 2022. 2, 3, 4

[23] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 1, 2, 3, 5

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv preprint arXiv:1512.03385*, 2015. 5, 6

[25] Xingyi He, Jiaming Sun, Yifan Wang, Sida Peng, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Detector-free structure from motion. In *arxiv*, 2023. 1, 2, 3, 4, 6, 7, 8

[26] Jared Heinly, Johannes L. Schonberger, Enrique Dunn, and Jan-Michael Frahm. Reconstructing the world* in six days *(as captured by the yahoo 100 million image dataset). In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015. 1

[27] Jakob Iglhaut, Carlos Cabo, Stefano Puliti, Livia Piermattei, James O'Connor, and Jacqueline Rosette. Structure from motion photogrammetry in forestry: A review. *Current Forestry Reports*, 5:155–168, 2019. 1

[28] Hanwen Jiang, Zhenyu Jiang, Kristen Grauman, and Yuke Zhu. Few-view object reconstruction with unknown categories and camera poses. *ArXiv*, 2212.04492, 2022. 1

[29] San Jiang, Cheng Jiang, and Wanshou Jiang. Efficient structure from motion for large-scale uav images: A review and a comparison of sfm tools. *ISPRS Journal of Photogrammetry and Remote Sensing*, 167:230–251, 2020. 1

[30] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6207–6217, 2021. 3

[31] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, 129(2):517–547, 2021. 2, 6, 7

[32] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Co-Tracker: It is better to track together. 2023. 2, 3, 4, 6

[33] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? *Proc. NeurIPS*, 2017. 4

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

[35] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018. 6

[36] Amy Lin, Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose++: Recovering 6d poses from sparse-view observations. *arXiv preprint arXiv:2305.04926*, 2023. 3, 5, 6, 7

[37] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 1

[38] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Proc. ECCV*, 2014. 6

[39] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-Perfect Structure-from-Motion with Featuremetric Refinement. *arXiv.cs*, abs/2108.08291, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[40] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. *arXiv preprint arXiv:2306.13643*, 2023. 3, 4, 6

[41] Yin Lou, Noah Snavely, and Johannes Gehrke. Matchminer: Efficient spanning structure mining in large image collections. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part II 12*, pages 45–58. Springer, 2012. 3

[42] David G. Lowe. Object Recognition from Local Scale-Invariant Features. In *Proc. ICCV*, 1999. 2

[43] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60(2), 2004. 2

[44] Xiao Xin Lu. A review of solutions for perspective-n-point problem in camera pose estimation. In *Journal of Physics: Conference Series*, page 052009. IOP Publishing, 2018. 3

[45] Wei-Chiu Ma, Anqi Joyce Yang, Shenlong Wang, Raquel Urtasun, and Antonio Torralba. Virtual correspondence: Humans as a cue for extreme-view geometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15924–15934, 2022. 3

[46] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *Proc. BMVC*, 2002. 2

[47] 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. *Proc. ECCV*, 2020. 5

[48] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1

[49] Jorge J Moré. The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis: proceedings of the biennial Conference held at Dundee, June 28–July 1, 1977*, pages 105–116. Springer, 2006. 5

[50] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Global fusion of relative motions for robust, accurate and scalable structure from motion. In *Proceedings of the IEEE international conference on computer vision*, pages 3248–3255, 2013. 2

[51] David Novotny, Diane Larlus, and Andrea Vedaldi. Learning 3d object categories by looking around them. In *Proc. ICCV*, 2017. 4

[52] John Oliensis. A critique of structure-from-motion algorithms. *Computer Vision and Image Understanding*, 80(2): 172–214, 2000. 2

[53] Onur Özyeşil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion*. *Acta Numerica*, 26:305–364, 2017. 1, 2

[54] Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky TQ Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, et al. Theseus: A library for differentiable nonlinear optimization. *Advances in Neural Information Processing Systems*, 35:3801–3818, 2022. 2, 5

[55] 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 IEEE/CVF International Conference on Computer Vision*, pages 10901–10911, 2021. 2, 6

[56] Peter Sand and Seth Teller. Particle video: Long-range motion estimation using point trajectories. *International journal of computer vision*, 80:72–91, 2008. 3

[57] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 2, 3, 4, 6

[58] Frederik Schaffalitzky and Andrew Zisserman. Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?". In *Proc. ECCV*, 2002. 1

[59] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2, 3, 4, 5

[60] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3260–3269, 2017. 2, 6, 8

[61] Yan Shi, Jun-Xiong Cai, Yoli Shavit, Tai-Jiang Mu, Wensen Feng, and Kai Zhang. Clustergnn: Cluster-based coarse-to-fine graph neural network for efficient feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12517–12526, 2022. 3

[62] Samarth Sinha, Jason Y Zhang, Andrea Tagliasacchi, Igor Gilitschenski, and David B Lindell. Sparsepose: Sparse-view camera pose regression and refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21349–21359, 2023. 3

[63] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 2, 5

[64] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 2, 3

[65] Chengzhou Tang and Ping Tan. Ba-net: Dense bundle adjustment network. *arXiv preprint arXiv:1806.04807*, 2018. 3

[66] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. *arXiv preprint arXiv:1812.04605*, 2018. 3

[67] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 3

[68] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems*, 34:16558–16569, 2021. 3

[69] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle Adjustment - A Modern Synthesis. In *Proc. ICCV Workshop*, 2000. 3

[70] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in Neural Information Processing Systems*, 33:14254–14265, 2020. 3, 4

[71] 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*, pages 5038–5047, 2017. 3

[72] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Proc. NeurIPS*, 2017. 2, 5

[73] Jianyuan Wang, Yiran Zhong, Yuchao Dai, Stan Birchfield, Kaihao Zhang, Nikolai Smolyanskiy, and Hongdong Li. Deep two-view structure-from-motion revisited. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 8953–8962, 2021. 3

[74] Jianyuan Wang, Christian Rupprecht, and David Novotny. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9773–9783, 2023. 3, 5, 6, 7

[75] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. In *Proceedings of the Asian Conference on Computer Vision*, pages 2746–2762, 2022. 2, 3

[76] 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. 1

[77] Tong Wei, Yash Patel, Alexander Shekhovtsov, Jiri Matas, and Daniel Barath. Generalized differentiable ransac. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17649–17660, 2023. 3

[78] Xingkui Wei, Yinda Zhang, Zhuwen Li, Yanwei Fu, and Xiangyang Xue. Deepsfm: Structure from motion via deep bundle adjustment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 230–247. Springer, 2020. 3, 7

[79] Matthew J Westoby, James Brasington, Niel F Glasser, Michael J Hambrey, and Jennifer M Reynolds. 'structure-from-motion'photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179:300–314, 2012. 1

[80] Kyle Wilson and Noah Snavely. Robust global translations with 1dsfm. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III 13*, pages 61–75. Springer, 2014. 2

[81] Changchang Wu. Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pages 127–134. IEEE, 2013. 2, 5

[82] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. Unsupervised learning of probably symmetric deformable 3d objects from images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1–10, 2020. 3

[83] Shangzhe Wu, Ruining Li, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. MagicPony: Learning articulated 3d animals in the wild. 2023. 3

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

[85] Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *ECCV*, pages 592–611. Springer, 2022. 3, 5, 6, 7

[86] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19855–19865, 2023. 3

[87] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017. 3