# RePoseD: Efficient Relative Pose Estimation With Known Depth Information

Yaqing Ding[1], Viktor Kocur[2], Václav Vávra[1], Zuzana Berger Haladová[2], Jian Yang[3], Torsten Sattler[4] and Zuzana Kukelova[1]
[1] Visual Recognition Group, Faculty of Electrical Engineering, Czech Technical University in Prague
[2] Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava
[3] PCA Lab, VCIP, College of Computer Science, Nankai University, Tianjin, China
[4] Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague

## Abstract

*Recent advances in monocular depth estimation methods (MDEs) and their improved accuracy open new possibilities for their applications. In this paper, we investigate how monocular depth estimates can be used for relative pose estimation. In particular, we are interested in answering the question whether using MDEs improves results over traditional point-based methods. We propose a novel framework for estimating the relative pose of two cameras from point correspondences with associated monocular depths. Since depth predictions are typically defined up to an unknown scale or even both unknown scale and shift parameters, our solvers jointly estimate the scale or both the scale and shift parameters along with the relative pose. We derive efficient solvers considering different types of depths for three camera configurations: (1) two calibrated cameras, (2) two cameras with an unknown shared focal length, and (3) two cameras with unknown different focal lengths. Our new solvers outperform state-of-the-art depth-aware solvers in terms of speed and accuracy. In extensive real experiments on multiple datasets and with various MDEs, we discuss which depth-aware solvers are preferable in which situation. The code is available at* https://github.com/kocurvik/mdrp.

## 1. Introduction

Relative pose estimation is a core problem in a wide range of computer vision applications, including Structure-from-Motion (SfM) [57], visual localization [55, 61, 71], and autonomous robots [43]. Typically, the relative pose of cameras is estimated from 2D-2D image point correspondences that are established using feature matching. Given 2D-2D matches, the relative pose [28, 37, 38, 51, 59, 60] of two cameras can be estimated using the epipolar geometry constraints [29]. For two calibrated cameras, 5 point correspondences are needed to estimate the 5 degrees of freedom of the relative pose (3 for rotation and two for the translation up to an unknown scale) [28, 37, 51, 60]. The relative pose of two cameras with unknown shared focal length can be estimated from 6 correspondences [28, 37, 38, 59]. If the full intrinsic calibration is unknown and/or both cameras have different calibrations, the relative pose can be estimated from 7 or 8 correspondences [29].

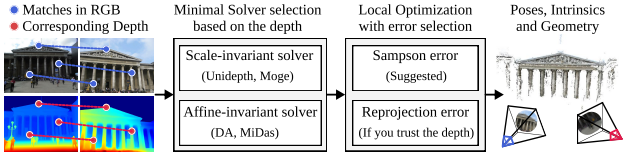


Figure 1. Given point matches and their corresponding depths from a pair of images, we propose the use of different minimal solvers for relative pose estimation, according to the properties of the depth data. For local optimization, the Sampson error is generally more robust across varying depth conditions, while the reprojection error may yield better results when the depth measurements are highly reliable.

Due to noise in the measurements and the presence of outlier correspondences, algorithms for relative pose estimation are usually employed in RANSAC-style hypothesize-and-verify frameworks [23]. The number of RANSAC iterations grows exponentially with the number of correspondences needed to estimate the pose. As such, a considerable amount of work focuses on reducing the number of correspondences necessary for pose estimation. This is typically done by using additional information, such as point coplanarity [10, 29], the gravity direction from an Inertial Measurement Unit (IMU) [15–17, 24, 36, 50, 62], or information about the local feature geometry such as the scale and rotation of features [3, 27, 46] or local affine frames [5, 6, 22].

Recently, learning-based methods [9, 26, 30, 48, 53, 64, 67, 68] have achieved significant success in predicting depth

from a single image. The increasing precision of Monocular Depth Estimation methods (MDEs) opens new possibilities of using depth information in downstream tasks. Recent studies have shown a growing focus on the use of dense depth estimates to improve the performance of various geometry-based applications, including structure-from-motion/SLAM [45, 49, 73], dense reconstruction [66, 70], and novel view synthesis [31, 42].

Depth naturally provides additional information for camera pose estimation. Thus, several recent works [2, 4, 20, 69] used monocular depths from MDEs to improve the relative pose estimation of cameras. The depth information provides additional constraints on the geometry, and thus reduces the number of correspondences necessary for the estimation. However, it also introduces new challenges: (1) Learning-based depth estimates are usually much more noisy and erroneous than image measurements. (2) MDEs typically produce scale/affine-invariant depth [9, 34, 48, 64], affine-invariant inverse depth [8, 54, 67, 68] or metric depth [7, 30, 53], meaning that the estimated depth or inverse depth is only defined up to an unknown scale factor or both unknown scale and shift parameters. (3) The unknown shift and scale can be different for different images of the same scene. Due to these challenges and the fact that the solvers proposed in [2, 20] use only relative depths, do not model unknown shifts, and have been tested only with depths from one network (Depth Anything [68] in [20], and a network for estimating the relative depth proposed directly in [2]), [2, 20] do not convincingly show the improvement of their depth-aware solvers over traditional 2D-2D point-based relative pose solvers [29, 51, 60]. The solvers proposed in [69], which model scale and shift, have been integrated into hybrid RANSAC [11] with Ceres-based optimization [1]. While this approach outperforms traditional methods, it is very time-consuming. Thus, prior work does not fully answer the question of whether recent MDEs provide useful information for the relative pose estimation task.

This paper attempts to provide an answer to this question by presenting a comprehensive comparison of different depth-aware solvers for calibrated, shared focal length and different focal lengths cases, under different conditions, *i.e.*, different sources of depth information, different feature detectors and matchers, different types of datasets (interior/exterior) and different optimization schemes inside RANSAC. To this end, we provide novel solvers for three different types of monocular depths: metric depth, scale-invariant depth, and affine-invariant depth. We introduce novel formulations that jointly estimate the relative pose of two cameras along with the scale or affine parameters of the monocular depths. The proposed solvers use the full information from the depth estimates, resulting in fewer correspondences necessary for the relative pose estimation compared to solvers based on relative depths [20].

The main contributions of the paper are:

- We integrate scale and affine parameters of monocular depth into solvers for relative pose estimation. These parameters are estimated together with the relative pose.
- We discuss all possible combinations of known and unknown shift and scale parameters. We propose several practical and efficient minimal solvers for calibrated cameras, two cameras with shared unknown focal length, and two cameras with different unkown focal lengths. Our solvers outperform state-of-the-art depth-aware solvers in terms of efficiency and accuracy.
- We evaluate different RANSAC schemes to account for potentially erroneous and noisy depth estimates.
- In extensive real experiments, we answer the questions which state-of-the-art MDEs are suitable for relative pose estimation, which require modeling unknown scale or affine parameters, and which provide the best improvement in terms of efficiency and accuracy over traditional, purely point correspondence-based solvers.

## 2. Related Work

Liwicki and Zach [46] show that the relative scales of SIFT [47] features can approximate the inverse relative depths of the features. In this case, one point correspondence provides two constraints for relative pose estimation, and the 5-DOF problem can be solved using only three point correspondences with two relative depths. In the same setting, Guan and Zhao [27] solve the generalized relative pose problem using three SIFT correspondences. Astermark *et al*. [2] propose a closed-form minimal solution to the relative pose of two calibrated pinhole cameras with known relative depths. In addition, they propose a neural network to estimate the relative depths. In [19], relative depths approximated from SIFT scales are used to estimate the calibrated homography. Barath and Sweeney [4] propose a 2-point solver with depth information for relative pose estimation. However, as pointed out in [69], this 2-point solver is degenerate for rigid alignments due to rank deficiency. In [20], the authors propose new minimal solvers that use relative depths for fundamental matrix estimation. They discuss different variants of solvers with known and unknown focal lengths. Their solvers work for monocular depths with unknown different scales. However, the solvers do not model unknown shifts in the depths, assuming that the shifts are zero in both images and thus can be ignored. All of the methods and minimal solvers discussed above use only the information from relative depths. Thus, they do not utilize the full information from the two monocular depth maps.

Dust3r [65] is a recent 3D foundation model for two-view geometry estimation tasks, including depth prediction and relative pose estimation. The relative pose is estimated by aligning two point maps via a similarity transformation.

Trained on large-scale datasets, Dust3r is able to handle scenarios where there is little overlap between the two images. Mast3r [41] further improves performance through feature matching. Mast3r minimizes both 3D-3D matching errors and 3D-2D reprojection errors to refine depths, relative poses, and intrinsic parameters.

In this paper, in contrast to the minimal solvers mentioned above that use only relative depths, we solve the problem using the full information from two depth maps. Concurrently to our work, Yu *et al*. [69] developed three novel minimal relative pose solvers for affine-invariant depths. Their calibrated, shared-focal, and two-focal length solvers solve the same formulation as our solvers with unknown shift and scales. In order to deal with potentially noisy and erroneous depth information, Yu *et al*. propose to use their depth-aware solvers jointly with classical 2D-2D point-based solvers in a hybrid LO-MSAC [11, 63] framework. In their Madpose framework, they combine the symmetric depth-induced reprojection error and the Sampson error for scoring pose hypotheses and for local optimization. The main differences between our work and [69] are the following: (1) Our solutions to configurations with unknown shifts and scales, *i.e*., affine-invariant depths, result in smaller and thus faster solvers. For a detailed comparison of sizes and run-times of the proposed and the MADPose solvers [69], see Table 2. (2) In addition to affine-invariant solvers, we also derive novel minimal solvers for cases with known/zero shifts and either unknown or known/same scales. Moreover, we derive a calibrated solver for inverse depth and discuss the number of solutions for all other variants of known/unknown/same/different shifts and scales (see Table 1). (3) By testing more variants of solvers, including novel solvers with zero shifts, we show that in some scenarios it is not necessary to model unknown shifts. This is in contrast to the observation made by [69], who, by missing some variants of the solvers in their experiments, concluded that modeling the unknown shift is always beneficial (even for "metric" depths). (4) In contrast to using a complex hybrid RANSAC scheme and a combination of reprojection and Sampson errors that results in slower running times, we look at a simple and more efficient local optimization scheme based on using a single solver with either reprojection or Sampson error.

## 3. Problem Statement

Assume that a set of 3D points $\Omega$ is observed by two cameras with projection matrices $\mathbf{K}_1[\mathbf{I} \mid \mathbf{0}]$ and $\mathbf{K}_2[\mathbf{R} \mid \mathbf{T}]$. Let $\{\mathbf{p}_i, \mathbf{q}_i\}, i = 1, \dots, n$ be a set of $n$ 2D point correspondences, *i.e*., the projections of the 3D points from $\Omega$ in the first and the second camera, respectively. Then we have

$$\lambda_i \mathbf{K}_2^{-1}\mathbf{q}_i = \eta_i \mathbf{R}\mathbf{K}_1^{-1}\mathbf{p}_i + \mathbf{T}, \tag{1}$$

where $\lambda_i$ and $\eta_i$ are the depths of the points $\mathbf{q}_i$ and $\mathbf{p}_i$.

MDEs provide either scale/affine-invariant depth [9, 34, 48, 64], affine-invariant inverse depth [8, 54, 67, 68], or metric depth [7, 30, 53]. In the most general case of affine-invariant depth, the true depths $\lambda_i$ and $\eta_i$ in (1) can be expressed as

$$\eta_i = s_1(\alpha_i + u), \ \lambda_i = s_2(\beta_i + v), \tag{2}$$

where $\alpha_i, \beta_i$ are estimated affine-invariant depths, and $\{s_1, s_2\}, \{u, v\}$ are the unknown scales and shifts of the depths. Substituting (2) into (1), we have

$$s_2(\beta_i + v)\mathbf{K}_2^{-1}\mathbf{q}_i = s_1(\alpha_i + u)\mathbf{R}\mathbf{K}_1^{-1}\mathbf{p}_i + \mathbf{T}, \tag{3}$$

Dividing (3) by $s_1$, results in

$$s(\beta_i + v)\mathbf{K}_2^{-1}\mathbf{q}_i = (\alpha_i + u)\mathbf{R}\mathbf{K}_1^{-1}\mathbf{p}_i + \mathbf{t}, \tag{4}$$

where $s = s_2/s_1$ and $\mathbf{t} = \mathbf{T}/s_1$, *i.e*., two different scales are replaced by the relative scale $s$. Note that in this case, the scale of the translation $\mathbf{t}$ is defined w.r.t. $s_1$, thus we can not further fix the scale of $\mathbf{t}$, *i.e*., the translation vector is not estimated up to scale like in the standard relative pose solvers [51], but needs to be estimated as a 3 degrees of freedom (3-DOF) vector. This means that for calibrated cameras, *i.e*., for known calibration matrices $\mathbf{K}_1$ and $\mathbf{K}_2$, we have a 9-DOF problem w.r.t. $\{s, u, v, \mathbf{R}, \mathbf{t}\}$. Since one 3D-3D point correspondence[1] provides three constraints of the form (4), we need at least 3 point correspondences with their monocular depths to solve this problem for the calibrated case. For the case of unknown focal lengths $f_1$ and $f_2$ in calibration matrices $\mathbf{K}_1$ and $\mathbf{K}_2$, we have 10 and 11 DoF, for cases $f_1 = f_2$ and $f_1 \neq f_2$ respectively.

In addition to affine-invariant depths, some MDE networks provide scale-invariant depths or even metric depths; moreover, in some applications the depth can be estimated directly using depth sensors. Thus in some scenarios, we can assume known/zero shift and/or equal scales and shifts of two depths. There are 6 possible combinations of known and unknown shifts and scales that result in different DOF problems. Table 1 lists all these combinations together with the number of solutions for different problem settings, with novel solvers to interesting practical setups that are tested in the main paper and in the Supplemental Material (SM) marked in gray. Note that for calibrated case the setups with known shifts and scales ($\{s, u, v\} = \{1, 0, 0\}$), and with known shifts and unknown scale $\{s, u, v\} = \{s, 0, 0\}$ can be solved using the well-known P3P solver [18, 25, 52]. The P3P solver can handle scale invariant depth, since it only uses single-side depth.[2] Due to space limits, we only

[1] Note that in this case the 2D points $\{\mathbf{p}_i, \mathbf{q}_i\}$ are equiped with depths and thus lifted to 3D points.

[2] Although one 3D-3D point correspondence provides three constraints, we can not use only two 3D-3D point correspondences to solve the 6-DOF pose problem with known $\{s, u, v\} = \{1, 0, 0\}$. The reason is that the rotation cannot be determined with only two points. There still remains a 1-DOF rotation around the line passing through the two points.

| | Solver | Point Correspondence | | | Scale Ratio | Shift | DOF | No. of Solutions |
|---|---|---|---|---|---|---|---|---|
| | | 3D-3D | 3D-2D | 2D-2D | | | | |
| Calibrated | 5-point [51] | - | - | 5 | - | - | 5 | 10 |
| | Rel3PT* [2] | 2 | - | 1 | 1 | (0, 0) | 5 | 4 |
| | P3P [18] | - | 3 | - | s | (0, -) | 6 | 4 |
| | 3PT$_{1uu}$ | 1 | 2 | 0 | 1 | (u, u) | 7 | 8 |
| | 3PT$_{1uv}$ | 2 | 1 | 0 | 1 | (u, v) | 8 | 8 |
| | 3PT$_{suu}$ | 2 | 1 | 0 | s | (u, u) | 8 | 8 |
| | 3PT$_{suv}$ | 3 | 0 | 0 | s | (u, v) | 9 | 4 |
| | 3PT$_{suv}$ (inverse) | 3 | 0 | 0 | s | (u, v) | 9 | 10 |
| Equal $f$ | 6-point [28] | - | - | 6 | - | - | 6 | 15 |
| | 3p3d* [20] | 3 | - | - | 1 | (0, 0) | 6 | 4 |
| | 3PT$_{100}f$ | 1 | 2 | 0 | 1 | (0, 0) | 7 | 6 |
| | 3PT$_{1uu}f$ | 2 | 1 | 0 | 1 | (u, u) | 8 | 6 |
| | 3PT$_{1uv}f$ | 3 | 0 | 0 | 1 | (u, v) | 9 | 4 |
| | 3PT$_{s00}f$ | 2 | 1 | 0 | s | (0, 0) | 8 | 4 |
| | 3PT$_{suu}f$ | 3 | 0 | 0 | s | (u, u) | 9 | 6 |
| | 4PT$_{suv}f$ | 3 | 0 | 1 | s | (u, v) | 10 | 8 |
| Varying $f_1, f_2$ | 7-point [29] | - | - | 7 | - | - | 7 | 3 |
| | 4p4d* [20] | 4 | - | - | s | (0, 0) | 7 | 1 |
| | 3PT$_{100}f_{1,2}$ | 2 | 1 | 0 | 1 | (0, 0) | 8 | 3 |
| | 3PT$_{1uu}f_{1,2}$ | 3 | 0 | 0 | 1 | (u, u) | 9 | 4 |
| | 4PT$_{1uv}f_{1,2}$ | 3 | 0 | 1 | 1 | (u, v) | 10 | 16 |
| | 3PT$_{s00}f_{1,2}$ | 3 | 0 | 0 | s | (0, 0) | 9 | 1 |
| | 4PT$_{suu}f_{1,2}$ | 3 | 0 | 1 | s | (u, u) | 10 | 18 |
| | 4PT$_{suv}f_{1,2}$ | 3 | 1 | 0 | s | (u, v) | 11 | 4 |

Table 1. Different combinations of using 3D-3D, 3D-2D, and 2D-2D point correspondences for focal length problems. The solvers marked gray are used in our experiments. Note that subscripts $suv$ in names of the solvers represent different configurations of known/unknown/same/different scales and shifts in the depth maps. *Relative depth was used, where one partial 3D-3D correspondence provides only 2 constraints.

discuss two practical scenarios in the main paper: the two depths up to unknown scale with two different shifts, and the two depth maps up to unknown scale. Other cases are discussed in the SM.

### 3.1. Calibrated Case

In this section, we first discuss the solver for the calibrated case, where the intrinsic parameters of cameras are known, *i.e.*, $\mathbf{K}_1$ and $\mathbf{K}_2$ are known. We use $\tilde{\mathbf{q}}_i = \mathbf{K}_2^{-1}\mathbf{q}_i$, $\tilde{\mathbf{p}}_i = \mathbf{K}_1^{-1}\mathbf{p}_i$ to represent the normalized image points.

Given 3 point correspondences and their monocular depths, substituting $\tilde{\mathbf{q}}_i, \tilde{\mathbf{p}}_i,\ i = 1, 2, 3$ into (4) we have nine equations in nine unknowns

$$s(\beta_i + v)\tilde{\mathbf{q}}_i = (\alpha_i + u)\mathbf{R}\tilde{\mathbf{p}}_i + \mathbf{t}. \tag{5}$$

By eliminating the translation from the above equations, *i.e.*, by subtracting pairs of equations, we have

$$s(\beta_i + v)\tilde{\mathbf{q}}_i - s(\beta_j + v)\tilde{\mathbf{q}}_j = \mathbf{R}((\alpha_i + u)\tilde{\mathbf{p}}_i - (\alpha_j + u)\tilde{\mathbf{p}}_j),$$

for $(i, j) = (1, 2), (1, 3), (2, 3)$.

Since applying the rotation matrix on a vector preserves the length of this vector, we have the following constraints

$$\|s(\beta_i + v)\tilde{\mathbf{q}}_i - s(\beta_j + v)\tilde{\mathbf{q}}_j\| = \|(\alpha_i + u)\tilde{\mathbf{p}}_i - (\alpha_j + u)\tilde{\mathbf{p}}_j\|,$$

| Solver | G-J | Eigen | Time ($\mu$s) |
|---|---|---|---|
| 3PT$_{suv}$(Ours) | $3 \times 6$ | Closed-form | 1.46 |
| 3PT$_{suv}$(M) | $12 \times 16$ | $4 \times 4$ | 4.45 |
| 4PT$_{suv}f$(Ours) | $24 \times 32$ | $8 \times 8$ | 12.5 |
| 4PT$_{suv}f$(M) | $36 \times 44$ | $8 \times 8$ | 23.6 |
| 4PT$_{suv}f_{1,2}$(Ours) | $20 \times 24$ | $4 \times 4$ | 6.45 |
| 4PT$_{suv}f_{1,2}$(M) | $40 \times 44$ | $4 \times 4$ | 20.2 |

Table 2. Efficiency comparison between the proposed affine-invariant solvers and those from MADPose [69], denoted as (M).

for $(i, j) = (1, 2), (1, 3), (2, 3)$. This is also known as the rigid motion constraint, which preserves the distance between two 3D points under a rigid transformation, and has been widely used in the literature [18, 23, 52, 72]. In this case, the rotation is eliminated and we obtain three equations in three unknowns. Since the equations only contain $s^2$, we can let $c = s^2$ to reduce the degrees and eliminate the symmetries. In short, if $s$ is a solution, then $-s$ is also a solution. However, $s$ should be positive since it presents the relative scale of two depths. thus a solution to $c = s^2$ gives us only one geometrically feasible solution for $s$.

The three above equations can be written in a matrix form $\mathbf{M}\,[cv^2, cv, c, u^2, u, 1]^\top = 0$, where $\mathbf{M}$ is a $3 \times 6$ coefficient matrix. After Gauss-Jordan (G-J) elimination of this matrix, the three monomials $\{cv^2, cv, c\}$ can be expressed as quadratic functions of $u$, *i.e.*, $cv^2 = g_1(u)$, $cv = g_2(u)$, $c = g_3(u)$, where $g_1, g_2, g_3$ are polynomials in $u$ of degree 2. Since $(cv)^2 = (c)(cv^2)$, we have $g_2^2 = g_1 g_3$, resulting in a quartic equation in $u$. This equation can be solved in closed form. We denote the final solver as 3PT$_{suv}$. Note that the inverse depth model results in a more complex solver without improving performance in our experiments (details provided in the SM). Therefore, we do not discuss the inverse depth model in the main paper.

### 3.2. Focal Length Problems

In many practical scenarios, the intrinsic parameters of the cameras may not be available. However, it is often reasonable to assume that modern cameras have square-shaped pixels, and the principal point coincides with the image center [28]. This is a widely used assumption in many camera geometry solvers, where the only unknown intrinsic parameters are focal lengths, *i.e.*, $\mathbf{K}_i = \mathrm{diag}(f_i, f_i, 1)$.

**Shared Unknown Focal Length**

First, let us assume that the two cameras have a shared unknown focal length, *i.e.*, $\mathbf{K}_1 = \mathbf{K}_2 = \mathrm{diag}(f, f, 1)$, which is a common scenario, *e.g.*, when estimating the motion of a single uncalibrated camera. This is a 10-DOF problem w.r.t. $\{s, u, v, \mathbf{R}, \mathbf{t}, f\}$, and we need at least three 3D-3D point correspondences and one 2D-2D point correspondence, *i.e.*, four 2D-2D correspondences from which three have monocular depths estimated in both images and

thus can be considered as 3D-3D correspondences. Note that, here the 3D-3D point correspondences depend on the unknown focal length parameter.

In this case, we may obtain the following system of 6 equations in 6 unknowns $\{s, u, v, f, \eta_4, \lambda_4\}$

$$\|s\mathbf{K}_2^{-1}((\beta_i + v)\mathbf{q}_i - s(\beta_j + v)\mathbf{q}_j)\| = \|\mathbf{K}_1^{-1}((\alpha_i + u)\mathbf{p}_i - (\alpha_j + u)\mathbf{p}_j)\|,$$
$$\|s\mathbf{K}_2^{-1}((\beta_k + v)\mathbf{q}_k - \lambda_4\mathbf{q}_4)\| = \|\mathbf{K}_1^{-1}((\alpha_k + u)\mathbf{p}_k - \eta_4\mathbf{p}_4)\|,$$

for $(i, j) = (1, 2), (1, 3), (2, 3)$ and $k = 1, 2, 3$. To solve such a system, we need to perform G-J elimination of a large matrix. However, the system has up to 28 real solutions. Thus, its solver is not practically relevant.

To find a more efficient and practical solution, we use four 2D-2D point correspondences with their four depths in both images. In this case, we obtain the following 6 equations in 4 unknowns $\{s, u, v, f\}$

$$\|s\mathbf{K}_2^{-1}((\beta_i + v)\mathbf{q}_i - (\beta_j + v)\mathbf{q}_j)\| = \|\mathbf{K}_1^{-1}((\alpha_i + u)\mathbf{p}_i - (\alpha_j + u)\mathbf{p}_j)\|,$$

where $i, j = 1, 2, 3, 4,\ i \neq j$. In general, it is an over-constrained system. One way to solve it is to use four of these six equations in four unknowns. The system of four polynomials can be solved using the Gröbner basis method [40], where the final solver performs G-J elimination of a $24 \times 32$ matrix and extracts up to 8 real solutions from the eigenvalues and eigenvectors of an $8 \times 8$ matrix. We denote this solver as 4PT$_{suv}f$. Alternatively, by using all six equations, we can derive a significantly smaller solver with up to two real solutions. This solver only needs to perform the G-J elimination of a $6 \times 8$ matrix. However, it is more sensitive to noise. Further details on this simplified focal length solver are provided in the SM.

With scale invariant depth, we have an 8-DOF problem w.r.t. $\{s, \mathbf{R}, \mathbf{t}, f\}$. We need at least two 3D-3D point correspondences and one 3D-2D point correspondence. The problem can be solved similarly as the calibrated case shown in Sec. 3.1. Due to the space limits, the details are shown in the SM, where we show a general solution to all problem configurations using three point correspondences.

**Different and Unknown Focal Lengths**

Finally, we consider the case where $\mathbf{K}_i = \text{diag}(f_i, f_i, 1),\ i = 1, 2$, with $f_1 \neq f_2$. This is a 11-DOF problem w.r.t. $\{s, u, v, \mathbf{R}, \mathbf{t}, f_1, f_2\}$. Thus, we need at least three 3D-3D point correspondences and one 3D-2D point correspondence, *i.e.*, a correspondence where we have monocular depth only in one image. Similarly to the equal and unknown focal length case, we can formulate this problem using a system of 6 equations in 6 unknowns with up to 14 real solutions and solve it using [40].

However, a more efficient solver can be obtained using four 3D-3D point correspondences, which yields six equations in five unknowns: $s, u, v, f_1, f_2$. This is an over-constrained system that can again be solved by using only five of these six equations. The five equations can be rewritten as $\mathbf{M}\,[1, c, f_1, uf_1, u^2f_1, cf_2, cvf_2, cv^2f_2]^\top = 0$, where $\mathbf{M}$ is a $5 \times 8$ coefficient matrix. Here, $f_2$ always appears together with $c$, thus, we let $cf_2 = \tilde{f}_2$ to simplify the polynomials. We obtain a solver that performs the G-J elimination of a $20 \times 24$ matrix and results in up to four solutions. We denote this solver as 4PT$_{suv}f_{1,2}$. Alternatively, we can use all six equations to derive a smaller solver, which is faster, but more sensitive to noise (see SM for more details).

With scale-invariant depth, we have a 9-DOF problem w.r.t. $\{s, \mathbf{R}, \mathbf{t}, f_1, f_2\}$. We need at least three 3D-3D point correspondences. This problem can be solved similarly as the calibrated case shown in Sec. 3.1. Due to space limits, the details on this 3PT$_{s00}f_{1,2}$ solver are shown in the SM.

## 4. Experiments

In this section, we present a comprehensive comparison of different depth-aware and point-based solvers for calibrated, shared focal length, and different focal length cases, under different conditions, *i.e.*, different sources of depth information, different feature detectors and matchers, different type of datasets (interior/exterior), and different optimization schemes inside RANSAC.

**Solvers.** (1) For calibrated relative pose estimation, we compare our 3PT$_{suv}$(ours) solver with the solver 3PT$_{suv}$(M) from [69], the Rel3PT solver [2], the P3P solver [18], and the 5PT point-based solver [51]. (2) For the unknown shared focal length case, we compared the proposed 3PT$_{s00}f$(ours) and 4PT$_{suv}f$(ours) solvers with the 4PT$_{suv}f$(M) solver from [69], the 3p3d solver [20], and the 6PT point-based solver [40]. (3) For the different focal length case, we compare our 3PT$_{s00}f_{1,2}$(ours) and 4PT$_{suv}f_{1,2}$(ours) solvers with the 4PT$_{suv}f_{1,2}$(M) solver from [69], the 4p4d solver [20], and the 7PT solver [29].

**Datasets.** In order to test the proposed solvers on real-world data, we choose the `Phototourism` [32], the `ETH3D` [58], and the `ScanNet` [13] datasets. The images of the `Phototourism` dataset were collected from multiple cameras obtained at different times, from different viewpoints, and with occlusions. It is a challenging outdoor dataset that is commonly used as a benchmark dataset [32] for camera geometry methods. We used five test scenes with $24,750$ image pairs from this dataset. The `ETH3D` multi-view stereo and 3D reconstruction benchmark [58] includes a range of indoor and outdoor scenes. Ground truth geometry was acquired using a high-precision laser scanner. Images were captured with a DSLR camera and a synchronized multi-camera rig featuring cameras with varying fields of view. In total, $4,144$ image pairs were used from the `ETH3D` dataset, and we used triangulation to obtain the ground truth depths. The `ScanNet` dataset is an indoor RGB-D video dataset containing 2.5 million views in more than 1,500 scans, an-

notated with camera poses and 3D surface reconstructions. We followed the evaluation setup in [56], and used 1,500 pairs from `ScanNet` for testing.

**Feature Detection and Matching.** We used three different image features, the first is the popular sparse SuperPoint (SP) features [14], widely used in SfM and localization pipelines [55]. For SuperPoint matching, we employ LightGlue (LG) [44], a state-of-the-art deep learning-based method that demonstrates significant improvements over traditional matching methods. The second is RoMa [21], a state-of-the-art dense matcher. The last is Mast3r [41], where feature matching is considered as a 3D task with DUSt3R [65], a recent and powerful 3D reconstruction framework based on Transformers.

**Depth Estimation.** We use five state-of-the-art MDE methods to obtain depth data, including Depth Anything v2 (DA v2) [68], UniDepth [53], MoGe [64], and Mast3r [41]. Experiments for MiDas [8] are provided in the SM. These MDE methods represent scale/affine-invariant [8, 41, 64, 68] as well as metric depths [53].

**Robust Estimation Frameworks.** We evaluated all solvers using the LO-RANSAC framework [12] within PoseLib [39] with either the Sampson (2 px threshold) or the reprojection error (16 px threshold) for LO and scoring, and a fixed number of 1,000 RANSAC iterations. In the main paper we show the results for the better scoring function for a given solver. Additional experiments are provided in the SM. We also evaluated selected solvers using the Hybrid RANSAC framework presented in [69] with 2 px Sampson error and 16 px reprojection error thresholds with equal weight for both types of error and fixed 1,000 iterations. For Mast3r [41], we use their non-linear optimization with 500 iterations for coarse refinement and 200 iterations for fine refinement. We observe that Mast3r performs better in our evaluation compared to the results reported in [69]. We attribute this potentially to using more refinement iterations, which in our experience improves performance.

**Evaluation Metrics.** In our experiments, we report the median pose error $\epsilon$ calculated as the maximum of rotation and translation errors in degrees. We also report the mean average accuracy (mAA) [32] with a threshold of 10 degrees. For the focal length solvers we use $f_{err} = |f_{est} - f_{gt}|/f_{gt}$ as the focal length error for one camera. When two cameras with different focal lengths are considered, we use the geometric mean of their errors. We report the median focal length errors denoted as $\epsilon_f$. We also report the mean average accuracy for focal lengths [35] at the 10% error threshold, denoted as $\text{mAA}_f$. Run-times $\tau$ are reported in milliseconds. Run-times for all methods except Mast3r are reported per one CPU of an Intel Xeon Gold 6338. The run-times of Mast3r are reported on the CPU of an Intel I7-11700KF with an RTX 3090 GPU. The run-times of all the methods do not include feature matching and depth estimation since they are equal for all depth-based solvers. For a fair comparison, for Mast3r, we only measure the run-times of the non-linear optimization, *i.e.*, time without feature matching, and initial depth estimation. For datasets with multiple scenes, we report the mean values across the scenes.

| Depth | Solver | Opt. | SP+LG [14, 44] | | | RoMA [21] | | |
|---|---|---|---|---|---|---|---|---|
| | | | $\epsilon(°)\downarrow$ | mAA$\uparrow$ | $\tau(ms)\downarrow$ | $\epsilon(°)\downarrow$ | mAA$\uparrow$ | $\tau(ms)\downarrow$ |
| - | 5PT [51] | S | 0.91 | 87.67 | 48.14 | 0.56 | 91.10 | 184.36 |
| Real Depth | Rel3PT [2] | S | 0.88 | 88.21 | 103.18 | 0.52 | 91.38 | 532.02 |
| | P3P [18] | S | 0.83 | 88.88 | 29.68 | 0.52 | 91.33 | 141.39 |
| | 3PT$_{suv}$(M) [69] | S | 0.79 | 88.55 | 41.81 | 0.45 | 91.39 | 145.59 |
| | 3PT$_{suv}$(**ours**) | S | 0.80 | 88.60 | **29.59** | 0.47 | 91.37 | **127.81** |
| | 3PT$_{suv}$(M) [69] | H [69] | **0.52** | 91.39 | 549.59 | **0.39** | **92.73** | 1505.19 |
| | 3PT$_{suv}$(**ours**) | H [69] | **0.52** | **91.42** | 543.48 | **0.39** | 92.72 | 1490.93 |
| DA v2 [68] | Rel3PT [2] | S | 5.57 | 68.46 | 35.74 | 3.90 | 80.56 | 145.85 |
| | P3P [18] | S | **0.90** | 86.25 | 23.26 | 0.72 | 90.61 | 93.99 |
| | 3PT$_{suv}$(M) [69] | S | **0.90** | **87.19** | 32.50 | 0.56 | 90.99 | 88.60 |
| | 3PT$_{suv}$(**ours**) | S | 0.91 | 87.07 | **21.68** | 0.56 | **91.01** | **75.87** |
| | 3PT$_{suv}$(M) [69] | H [69] | 0.97 | 85.52 | 605.63 | **0.54** | 90.62 | 1493.75 |
| | 3PT$_{suv}$(**ours**) | H [69] | 0.98 | 85.56 | 593.95 | **0.54** | 90.62 | 1477.73 |
| MoGe [64] | Rel3PT [2] | S | 4.74 | 72.08 | 42.19 | 2.74 | 82.04 | 170.29 |
| | P3P [18] | S | 0.91 | 87.67 | 25.72 | 0.54 | 91.16 | 111.74 |
| | 3PT$_{suv}$(M) [69] | S | 0.89 | 87.71 | 33.45 | 0.53 | 91.04 | 98.20 |
| | 3PT$_{suv}$(**ours**) | S | 0.89 | 87.67 | **22.41** | 0.54 | 91.05 | **84.24** |
| | 3PT$_{suv}$(M) [69] | H [69] | 0.86 | **88.26** | 566.16 | 0.50 | 91.17 | 1414.96 |
| | 3PT$_{suv}$(**ours**) | H [69] | **0.85** | 88.24 | 554.79 | **0.49** | **91.23** | 1401.61 |
| UniDepth [53] | Rel3PT [2] | S | 1.36 | 78.82 | 49.70 | 0.70 | 88.25 | 207.86 |
| | P3P [18] | S | 0.88 | 88.00 | 25.93 | 0.56 | 91.11 | 112.65 |
| | 3PT$_{suv}$(M) [69] | S | 0.94 | 87.42 | 33.90 | 0.55 | 91.04 | 97.85 |
| | 3PT$_{suv}$(**ours**) | S | 0.95 | 87.49 | **22.58** | 0.55 | 91.01 | **83.72** |
| | 3PT$_{suv}$(M) [69] | H [69] | **0.86** | **88.03** | 558.61 | **0.53** | **91.33** | 1402.46 |
| | 3PT$_{suv}$(**ours**) | H [69] | **0.86** | **88.03** | 550.74 | **0.53** | **91.33** | 1392.71 |

| Depth | Solver | Opt. | Mast3r [41] | | |
|---|---|---|---|---|---|
| | | | $\epsilon(°)\downarrow$ | mAA$\uparrow$ | $\tau(ms)\downarrow$ |
| - | 5PT [51] | S | **0.66** | 90.29 | 126.77 |
| Mast3r [41] | Rel3PT [2] | S | 0.67 | 90.30 | 104.46 |
| | P3P [18] | S | 0.67 | 90.24 | 56.52 |
| | 3PT$_{suv}$(M) [69] | S | 0.67 | 90.20 | 42.49 |
| | 3PT$_{suv}$(**ours**) | S | 0.67 | **90.35** | **30.90** |
| | 3PT$_{suv}$(M) [69] | H [69] | 0.92 | 87.96 | 2647.02 |
| | 3PT$_{suv}$(**ours**) | H [69] | 0.92 | 87.98 | 2635.27 |

Table 3. Comparison of different methods on the `ETH3D` dataset [58] for the calibrated case. Opt.: S - PoseLib [39] implementation using Sampson error, H - hybrid RANSAC from [69].

## 4.1. Calibrated Relative Pose Estimation

Table 3 shows the comparison of different methods on the `ETH3D` dataset. The results for the `Phototourism` and `ScanNet` datasets are provided in the SM. The results show that using depth in pose estimation results in more accurate poses when good depth estimates (e.g. MoGe [64], UniDepth [53]) are available. In this case, the overall best-performing methods use the hybrid RANSAC proposed in [69], with our solver 3PT$_{suv}$(ours) slightly outperforming 3PT$_{suv}$(M) [69]. We note that the use of the hybrid RANSAC scheme comes at a significant increase in computational cost over the use of PoseLib [39]. As the authors of [69] point out, their implementation of the hybrid RANSAC scheme, which we use in our experiments, could be optimized for computational efficiency. However, even with optimizations the method requires significantly more computation per one iteration than standard RANSAC.

Based on the results for PoseLib [39], when good depth

is available, P3P [18] and 3PT$_{suv}$ have similar performance. When less accurate depth estimates are available, 5PT performs best especially with good matches (*e.g.*, Mast3r [41], RoMA [21]).

### 4.2. Shared Unknown Focal Length

Table 4 shows the comparison of the methods on the ETH3D dataset [58] for two cameras with a shared unknown focal length. When a good depth estimate is available, the best results are obtained utilizing the hybrid RANSAC scheme from [69]. We show that our solver 3PT$_{s00}f$ outperforms 4PT$_{suv}f$(M) [69] for many combinations of MDEs and matches despite not modeling shift. Our 3PT$_{s00}f$ solver also performs best when comparing solvers incorporated within PoseLib [39] and used in combination with MoGe [64], UniDepth [53], and Mast3r [41]. For DA v2 [68] and MiDas [8] (results provided in SM) modeling shift seems to be beneficial, and we show that our 4PT$_{suv}f$ solver performs on par with 4PT$_{suv}f$(M) while being 2x faster. Our observation that modeling an unknown shift is not always necessary is in contradiction to the observation made in [69]. The reason is that authors in [69] have not tested a scale-invariant 3PT solver and have not incorporated other depth-aware solvers into the hybrid RANSAC scheme. We also note that despite performing expensive non-linear optimization to obtain the poses, Mast3r [41] performs worse than approaches based on RANSAC when either good depth (UniDepth [53], MoGe [64]) or matches (RoMA [21]) are available.

A comparison of the speed-accuracy trade-off of various methods is shown in Figure 2 for the combination of RoMA [21] and UniDepth [53]. The figure shows that when fast computation is necessary, our solver 3PT$_{s00}f$ implemented in PoseLib [39] performs the best. However, in general, the best accuracy is achieved using the same solver within the hybdrid RANSAC scheme from [69].

| Matches | Depth | Solver | Opt | ETH | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | $\epsilon(^\circ)\downarrow$ | $\epsilon_f(^\circ)\downarrow$ | mAA$\uparrow$ | mAA$_f\uparrow$ | $\tau(ms)\downarrow$ |
| SP+LG [14, 44] | - | 6PT [40] | S | 2.45 | 0.04 | 75.57 | 61.52 | 80.02 |
| | Real Depth | 3p3d [20] | S | 2.06 | 0.04 | 78.00 | 62.83 | 30.70 |
| | | 4PT$_{suv}f$(M) [69] | S | 1.83 | 0.03 | 78.86 | 63.71 | 112.34 |
| | | 4PT$_{suv}f$(**ours**) | S | 1.72 | 0.03 | 78.90 | 63.56 | 51.10 |
| | | 3PT$_{s00}f$(**ours**) | S | 1.75 | 0.03 | 79.17 | 63.63 | **25.11** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | **1.07** | **0.02** | 82.01 | 75.63 | 1502.02 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | **1.07** | **0.02** | **82.19** | **75.76** | 1411.05 |
| | DA v2 [68] | 3p3d [20] | S | 4.54 | 0.08 | 61.60 | 48.76 | 27.01 |
| | | 4PT$_{suv}f$(M) [69] | S | 2.10 | **0.04** | 74.60 | 60.77 | 104.75 |
| | | 4PT$_{suv}f$(**ours**) | S | **2.02** | **0.04** | **74.67** | 60.41 | 46.62 |
| | | 3PT$_{s00}f$(**ours**) | S | 2.20 | **0.04** | 73.66 | **60.93** | **21.62** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | 2.34 | 0.05 | 70.07 | 58.00 | 1190.89 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | 2.33 | 0.05 | 70.16 | 58.31 | 1216.89 |
| | MoGe [64] | 3p3d [20] | S | 3.18 | 0.06 | 68.24 | 54.64 | 27.70 |
| | | 4PT$_{suv}f$(M) [69] | S | 2.10 | 0.04 | 76.06 | 61.81 | 107.30 |
| | | 4PT$_{suv}f$(**ours**) | S | 2.15 | 0.04 | 75.59 | 60.88 | 57.91 |
| | | 3PT$_{s00}f$(**ours**) | S | 1.99 | 0.04 | 76.94 | 62.66 | **24.72** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | 1.50 | **0.03** | 79.23 | 66.34 | 956.33 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | **1.41** | **0.03** | **80.24** | **67.42** | 967.52 |
| | UniDepth [53] | 3p3d [20] | S | 3.49 | 0.07 | 69.47 | 55.57 | 27.57 |
| | | 4PT$_{suv}f$(M) [69] | S | 2.21 | 0.04 | 75.61 | 61.37 | 106.73 |
| | | 4PT$_{suv}f$(**ours**) | S | 1.92 | 0.04 | 76.18 | 62.00 | 46.97 |
| | | 3PT$_{s00}f$(**ours**) | S | 2.04 | 0.04 | 76.89 | 62.42 | **23.47** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | **1.27** | **0.03** | 81.68 | **69.64** | 1107.04 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | 1.28 | **0.03** | **81.99** | 69.28 | 1149.25 |
| RoMA [21] | - | 6PT [40] | S | 1.15 | 0.02 | 85.23 | 75.03 | 147.48 |
| | Real Depth | 3p3d [20] | S | **0.93** | **0.02** | 85.98 | 74.98 | 113.30 |
| | | 4PT$_{suv}f$(M) [69] | S | 0.99 | **0.02** | 86.37 | **75.21** | 167.49 |
| | | 4PT$_{suv}f$(**ours**) | S | 1.03 | **0.02** | 86.25 | 75.04 | 107.05 |
| | | 3PT$_{s00}f$(**ours**) | S | 0.99 | **0.02** | **86.68** | 74.99 | **79.96** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | 2.12 | **0.02** | 76.69 | 72.51 | 3218.22 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | 2.28 | **0.02** | 76.42 | 72.43 | 3067.57 |
| | DA v2 [68] | 3p3d [20] | S | 2.07 | **0.02** | 78.86 | 68.29 | 91.39 |
| | | 4PT$_{suv}f$(M) [69] | S | 1.21 | **0.02** | 85.33 | 75.09 | 146.35 |
| | | 4PT$_{suv}f$(**ours**) | S | **1.20** | **0.02** | **85.61** | **75.43** | 88.03 |
| | | 3PT$_{s00}f$(**ours**) | S | 1.23 | **0.02** | 84.89 | 75.00 | **67.61** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | 1.28 | 0.03 | 81.56 | 68.72 | 2161.36 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | 1.30 | 0.03 | 81.57 | 68.68 | 2245.95 |
| | MoGe [64] | 3p3d [20] | S | 1.56 | **0.02** | 80.98 | 70.19 | 98.44 |
| | | 4PT$_{suv}f$(M) [69] | S | 1.02 | **0.02** | 85.91 | 75.61 | 151.65 |
| | | 4PT$_{suv}f$(**ours**) | S | 1.10 | **0.02** | 85.85 | 75.80 | 92.92 |
| | | 3PT$_{s00}f$(**ours**) | S | 1.05 | **0.02** | 86.04 | 75.83 | **75.12** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | **0.89** | **0.02** | 86.99 | **76.66** | 1923.31 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | 0.91 | **0.02** | **87.20** | 76.50 | 2043.91 |
| | UniDepth [53] | 3p3d [20] | S | 1.89 | **0.02** | 82.50 | 71.89 | 97.99 |
| | | 4PT$_{suv}f$(M) [69] | S | 1.12 | **0.02** | 85.63 | 75.66 | 151.31 |
| | | 4PT$_{suv}f$(**ours**) | S | 1.12 | **0.02** | 85.59 | 75.64 | 91.92 |
| | | 3PT$_{s00}f$(**ours**) | S | 1.04 | **0.02** | 85.78 | 75.69 | **75.62** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | 0.83 | **0.02** | 87.30 | 77.34 | 1997.88 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | **0.82** | **0.02** | **87.53** | **77.79** | 2133.46 |
| Mast3r [41] | - | 6PT [40] | S | 1.23 | 0.03 | 82.99 | 69.00 | 85.89 |
| | Mast3r [41] | 3p3d [20] | S | 1.37 | 0.03 | 81.30 | 67.31 | 38.96 |
| | | 4PT$_{suv}f$(M) [69] | S | 1.58 | 0.03 | 79.68 | 66.05 | 95.89 |
| | | 4PT$_{suv}f$(**ours**) | S | 1.36 | 0.03 | 80.86 | 67.16 | 48.36 |
| | | 3PT$_{s00}f$(**ours**) | S | 1.34 | 0.03 | 82.09 | 68.26 | **34.54** |
| | | 4PT$_{suv}f$(M) [69] | H [69] | 2.43 | 0.05 | 72.28 | 58.32 | 3825.11 |
| | | 3PT$_{s00}f$(**ours**) | H [69] | 2.39 | 0.04 | 72.45 | 58.61 | 4079.21 |
| | | - | M [41] | **1.32** | **0.01** | **85.64** | **82.95** | 4800.37 |

Table 4. Comparison of different methods on the ETH3D dataset [58] for the equal and unknown focal length case. Opt.: S - PoseLib [39] implementation using Sampson error, H - hybrid RANSAC from [69], M - non-linear optimization used in [41].

### 4.3. Different Focal Lengths

Table 5 shows the comparison of the methods on the Phototourism dataset for two cameras with different unknown focal lengths. As for the previous two cases, when good depth estimates are available the best results are obtained with the hybrid RANSAC scheme [69] in which our solver 3PT$_{s00}f_{1,2}$ outperforms 4PT$_{suv}f_{1,2}$(M) [69], showing that it is not necessary to model shift for good performance. This is also confirmed by the experiments in PoseLib [39] which show that 3PT$_{s00}f$ outperforms other depth-based solvers across all combinations of matches and MDEs in terms of pose accuracy while also being significantly faster than solvers which incorporate shift. Figure 3 shows that 3PT$_{s00}f$ clearly outperforms the alternatives in terms of speed and pose estimation accuracy. The results for this case also show that when good depth estimates are available it is possible to obtain significantly better results by using reprojection errors. This is due to the fact the the problem has a very commonly occurring degeneracy when only point correspondences are used [33].

**Additional Experiments.** We provide additional experiments in the SM, which, in addition to results on additional datasets, also provide an evaluation of alternative scoring and LO strategies within PoseLib, as well as the proposed 3PT$_{100}f$ and 3PT$_{100}f_{1,2}$ solvers. We also show that it is possible to use Mast3r [41] for matches in combination with MDE to improve estimated poses and intrinsics.

**Limitations.** Although using depth maps does not add computational overhead in pose estimation, obtaining these depth maps using learning-based methods introduces additional costs. This is usually a few milliseconds per im-

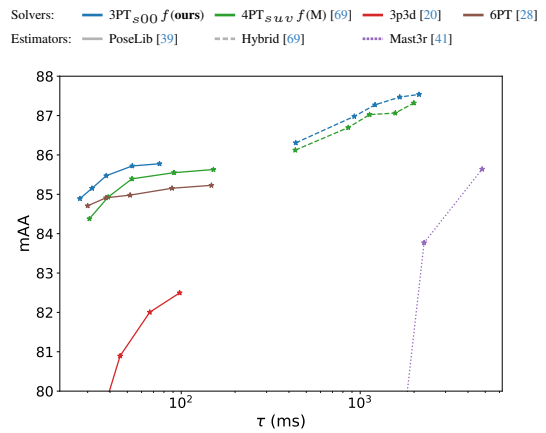


Figure 2. Speed-accuracy evaluation for the case of two cameras with shared unknown focal length using RoMA matches [21] and UniDepth [53] depths on the `ETH3D` dataset [58]. We evaluated mAA and runtimes ($\tau$) for each method running for 50, 100, 200, 500, and 1,000 iterations, with the exception of Mast3r [41], which ran with a maximum of 500 iterations.

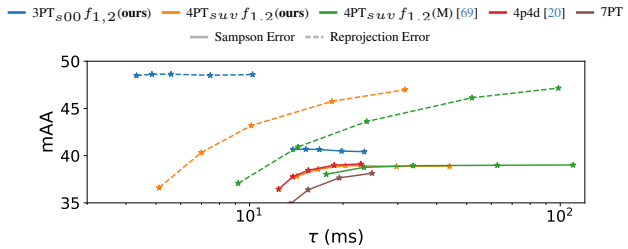


Figure 3. Speed-accuracy evaluation for the different unknown focal lengths case using SP+LG matches [14, 44] and MoGe [64] depths on the `Phototourism` dataset [32]. We evaluated mAA and runtimes ($\tau$) for each method running for 50, 100, 200, 500, and 1,000 iterations within PoseLib [39].

age [69]. However, in many applications, MDE methods are used for multiple tasks. Thus, the computational overhead is not solely attributable to the pose estimation method.

Compared to standard point-based focal length solvers, the proposed solvers encounter fewer degenerate cases. The formulation allows us to handle pure rotation effectively. However, there are still several degenerate cases similar to those in point-based solvers; for example, four coplanar points and cases of pure translation can introduce degeneracies in focal length recovery [33].

## 5. Conclusion

We address the problem of estimating the relative pose of two cameras using monocular depth predictions. We propose novel solvers that jointly estimate the scale and shift (or only one of them), and the relative pose. We consider solvers for calibrated cameras, and cameras with shared unknown or different unknown focal lengths. Efficient solvers that outperform state-of-the-art depth-aware solvers are proposed for all three cases. In extensive experiments, we discuss which solvers are preferable in different situations, *e.g.*, for precise or imprecise depths. We find that when depth estimates are accurate, modeling only scale tends to yield better performance. In such cases, it's also more effective to use reprojection error for the focal length problems.

| Matches | Depth | Solver | Opt | Phototourism | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | $\epsilon(^\circ)\downarrow$ | $\epsilon_f(^\circ)\downarrow$ | mAA↑ | mAA$_f$ ↑ | $\tau(ms)\downarrow$ |
| SP+LG [14, 44] | - | 7PT [29] | S | 8.03 | 0.17 | 38.14 | 23.78 | 24.80 |
| | Real Depth | 4p4d [20] | S | 7.58 | 0.16 | 39.34 | 24.65 | 23.69 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | R | 2.27 | **0.03** | 69.17 | 63.33 | 106.32 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | R | 2.28 | **0.03** | 69.14 | 63.31 | 24.35 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | R | **2.23** | **0.03** | **69.50** | **63.66** | **12.05** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 2.86 | 0.04 | 63.88 | 53.46 | 1922.12 |
| | | 4p4d [20] | H [69] | 2.89 | 0.05 | 63.21 | 52.87 | 1921.03 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | 2.82 | 0.04 | 64.19 | 53.73 | 1901.23 |
| | DA v2 [68] | 4p4d [20] | S | 10.40 | 0.20 | 34.72 | 22.08 | 22.02 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | S | 8.17 | 0.18 | 37.62 | 23.32 | 113.20 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | S | 8.19 | 0.18 | 37.62 | 23.27 | 33.14 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | S | **7.66** | **0.16** | **38.90** | **23.57** | **20.44** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 9.55 | 0.21 | 32.88 | 20.88 | 2490.39 |
| | | 4p4d [20] | H [69] | 9.89 | 0.22 | 32.14 | 20.41 | 2402.27 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | 9.48 | 0.21 | 32.49 | 20.65 | 2494.66 |
| | MoGe [64] | 4p4d [20] | S | 7.71 | 0.16 | 39.12 | 24.44 | 22.82 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | R | 6.60 | 0.08 | 47.16 | 39.06 | 98.69 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | R | 6.71 | 0.08 | 46.98 | 38.96 | 31.64 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | R | 5.73 | **0.07** | 48.59 | 40.41 | **10.22** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 3.88 | **0.07** | 57.21 | 42.80 | 1953.23 |
| | | 4p4d [20] | H [69] | 4.19 | **0.07** | 55.76 | 41.76 | 1881.29 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | **3.83** | **0.07** | **57.62** | **43.25** | 2031.28 |
| | UniDepth [53] | 4p4d [20] | S | 7.67 | 0.16 | 39.18 | 24.61 | 23.63 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | R | 4.90 | 0.07 | 52.64 | 43.12 | 106.30 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | R | 4.89 | 0.07 | 52.59 | 43.12 | 24.46 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | R | 4.82 | 0.07 | 52.85 | 43.28 | **12.89** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 3.38 | **0.06** | 60.15 | 47.02 | 1945.50 |
| | | 4p4d [20] | H [69] | 3.45 | **0.06** | 59.49 | 46.63 | 1940.62 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | **3.35** | **0.06** | **60.38** | **47.24** | 1957.65 |
| RoMA [21] | - | 7PT [29] | S | 4.30 | 0.10 | 53.16 | 34.73 | 75.10 |
| | Real Depth | 4p4d [20] | S | 4.24 | 0.10 | 53.51 | 35.01 | 77.49 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | R | 1.25 | **0.02** | 79.75 | 73.91 | 136.12 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | R | 1.26 | **0.02** | 79.68 | 73.84 | 54.16 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | R | **1.21** | **0.02** | **80.37** | **74.49** | **48.02** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 1.60 | 0.03 | 75.10 | 64.07 | 3518.89 |
| | | 4p4d [20] | H [69] | 1.63 | 0.03 | 74.63 | 63.62 | 3199.77 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | 1.58 | 0.03 | 75.41 | 64.37 | 3531.28 |
| | DA v2 [68] | 4p4d [20] | S | 4.61 | 0.11 | 51.24 | 33.30 | **73.93** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | S | 4.43 | **0.10** | 52.90 | 34.36 | 170.68 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | S | 4.42 | **0.10** | 52.83 | 34.35 | 90.36 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | S | **4.20** | **0.10** | **53.95** | **34.56** | 81.59 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 6.53 | 0.16 | 43.25 | 27.71 | 4026.79 |
| | | 4p4d [20] | H [69] | 6.56 | 0.16 | 43.21 | 27.74 | 3676.08 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | 6.52 | 0.16 | 43.30 | 27.79 | 4113.61 |
| | MoGe [64] | 4p4d [20] | S | 4.22 | 0.10 | 53.61 | 34.99 | 77.14 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | R | 4.87 | 0.07 | 54.21 | 43.35 | 128.82 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | R | 4.90 | 0.07 | 54.05 | 43.34 | 48.16 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | R | 4.40 | 0.06 | 55.38 | 44.33 | **46.38** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 2.41 | **0.05** | 69.08 | 52.39 | 3663.43 |
| | | 4p4d [20] | H [69] | 2.44 | **0.05** | 68.75 | 52.18 | 3330.63 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | **2.39** | **0.05** | **69.29** | **52.57** | 3803.60 |
| | UniDepth [53] | 4p4d [20] | S | 4.20 | 0.10 | 53.84 | 35.24 | 77.78 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | R | 4.20 | 0.06 | 56.76 | 47.14 | 135.10 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | R | 4.20 | 0.06 | 56.72 | 47.10 | **53.40** |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | R | 4.06 | 0.06 | 57.16 | 47.32 | 54.73 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 2.30 | **0.04** | 69.95 | 53.90 | 3440.95 |
| | | 4p4d [20] | H [69] | 2.31 | **0.04** | 69.67 | 53.64 | 3114.21 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | **2.29** | **0.04** | **70.18** | **54.09** | 3533.26 |
| Mast3r [41] | - | 7PT [29] | S | 4.39 | 0.09 | 54.01 | 35.02 | 39.53 |
| | Mast3r [41] | 4p4d [20] | S | 5.20 | 0.11 | 49.69 | 31.64 | **36.90** |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | S | 7.02 | 0.14 | 43.95 | 27.62 | 108.64 |
| | | 4PT$_{suv}f_{1,2}$(**ours**) | S | 7.03 | 0.14 | 44.10 | 27.81 | 44.88 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | S | 5.40 | 0.12 | 49.25 | 30.88 | 37.40 |
| | | 4PT$_{suv}f_{1,2}$(M) [69] | H [69] | 10.11 | 0.24 | 35.16 | 20.56 | 4871.24 |
| | | 4p4d [20] | H [69] | 32.03 | 0.76 | 1.33 | 2.49 | 6492.27 |
| | | 3PT$_{s00}f_{1,2}$(**ours**) | H [69] | 10.13 | 0.25 | 35.17 | 20.51 | 4980.42 |
| | | - | M [41] | **2.71** | **0.04** | **66.54** | **56.43** | 4903.10 |

Table 5. Comparison of different methods on the `Phototourism` dataset [32] for the two unknown focal lengths case. Opt.: S/R - PoseLib [39] implementation using Sampson error (S) and reprojection error (R), H - hybrid RANSAC from [69], M - non-linear optimization used in [41].

**Acknowledgments.** This work was supported by the Czech Science Foundation (GACR) JUNIOR STAR, Grant No. 22-23183M, the Student Grant Competition of the Czech Technical University in Prague, Grant No. SGS23/173/OHK3/3T/13, the EU NextGenerationEU through the Recovery and Resilience Plan for Slovakia under the project "InnovAIte Slovakia, Illuminating Pathways for AI-Driven Breakthroughs" No. 09I02-03-V01-00029, and the NSFC under Grant Nos. 62361166670 and U24A20330. The access to the computational infrastructure of the OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics" is also gratefully acknowledged. Part of the results was obtained using the computational resources procured in the project National competence centre for high performance computing (project code: 311070AKF2) funded by European Regional Development Fund, EU Structural Funds Informatization of society, Operational Program Integrated Infrastructure.

## References

[1] Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres Solver, 2023. 2

[2] Jonathan Astermark, Yaqing Ding, Viktor Larsson, and Anders Heyden. Fast relative pose estimation using relative depth. In *International Conference on 3D Vision (3DV)*, 2024. 2, 4, 5, 6

[3] Daniel Barath and Zuzana Kukelova. Relative pose from sift features. In *European Conference on Computer Vision*, pages 454–469. Springer, 2022. 1

[4] Daniel Barath and Chris Sweeney. Relative pose solvers using monocular depth. In *2022 26th International Conference on Pattern Recognition (ICPR)*, 2022. 2

[5] Daniel Barath, Tekla Toth, and Levente Hajder. A minimal solution for two-view focal-length estimation using two affine correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1

[6] Jacob Bentolila and Joseph M Francos. Conic epipolar constraints from affine correspondences. *Computer Vision and Image Understanding*, 2014. 1

[7] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 2, 3

[8] Reiner Birkl, Diana Wofk, and Matthias Müller. Midas v3. 1–a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*, 2023. 2, 3, 6, 7

[9] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 1, 2, 3

[10] Shen Cai, Zhanhao Wu, Lingxi Guo, Jiachun Wang, Siyu Zhang, Junchi Yan, and Shuhan Shen. Fast and interpretable 2d homography decomposition: Similarity-kernel-similarity and affine-core-affine transformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 1

[11] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid Camera Pose Estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 3

[12] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Pattern Recognition: 25th DAGM Symposium, Magdeburg, Germany, September 10-12, 2003. Proceedings 25*, pages 236–243. Springer, 2003. 6

[13] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017. 5

[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*, 2018. 6, 7, 8

[15] Yaqing Ding, Jian Yang, Jean Ponce, and Hui Kong. An efficient solution to the homography-based relative pose problem with a common reference direction. In *International Conference on Computer Vision (ICCV)*, 2019. 1

[16] Yaqing Ding, Daniel Barath, and Zuzana Kukelova. Minimal solutions for panoramic stitching given gravity prior. In *International Conference on Computer Vision (ICCV)*, pages 5579–5588, 2021.

[17] Yaqing Ding, Jian Yang, Jean Ponce, and Hui Kong. Homography-based minimal-case relative pose estimation with known gravity direction. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2022. 1

[18] Yaqing Ding, Jian Yang, Viktor Larsson, Carl Olsson, and Kalle Åström. Revisiting the p3p problem. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 3, 4, 5, 6, 7

[19] Yaqing Ding, Jonathan Astermark, Magnus Oskarsson, and Viktor Larsson. Noisy one-point homographies are surprisingly good. In *Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[20] Yaqing Ding, Václav Vávra, Snehal Bhayani, Qianliang Wu, Jian Yang, and Zuzana Kukelova. Fundamental matrix estimation using relative depths. In *European Conference on Computer Vision (ECCV)*, 2024. 2, 4, 5, 7, 8

[21] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 6, 7, 8

[22] Ivan Eichhardt and Daniel Barath. Relative pose from deep learned depth and a single affine correspondence. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[23] 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*, 1981. 1, 4

[24] Friedrich Fraundorfer, Petri Tanskanen, and Marc Pollefeys. A minimal case solution to the calibrated relative pose problem for the case of two known orientation angles. In *European Conference on Computer Vision (ECCV)*, 2010. 1

[25] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 2003. 3

[26] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *International Conference on Computer Vision (ICCV)*, 2019. 1

[27] Banglei Guan and Ji Zhao. Relative pose estimation for multi-camera systems from point correspondences with scale ratio. In *Proceedings of the 30th ACM International Conference on Multimedia*, 2022. 1, 2

[28] Richard Hartley and Hongdong Li. An efficient hidden variable approach to minimal-case camera motion estimation. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2012. 1, 4, 8

[29] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 1, 2, 4, 5, 8

[30] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *arXiv preprint arXiv:2404.15506*, 2024. 1, 2, 3

[31] Kaiwen Jiang, Yang Fu, Mukund Varma T, Yash Belhe, Xiaolong Wang, Hao Su, and Ravi Ramamoorthi. A construct-optimize approach to sparse view synthesis without camera pose. In *ACM SIGGRAPH 2024 Conference Papers*, 2024. 2

[32] 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*, 2020. 5, 6, 8

[33] Fredrik Kahl and Bill Triggs. Critical motions in euclidean structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, 1999. 7, 8

[34] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 3

[35] Viktor Kocur, Daniel Kyselica, and Zuzana Kukelova. Robust self-calibration of focal lengths from the fundamental matrix. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5220–5229, 2024. 6

[36] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Closed-form solutions to minimal absolute pose problems with known vertical direction. In *Asian Conference on Computer Vision (ACCV)*, 2010. 1

[37] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Polynomial eigenvalue solutions to minimal problems in computer vision. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2012. 1

[38] Zuzana Kukelova, Joe Kileel, Bernd Sturmfels, and Tomas Pajdla. A clever elimination strategy for efficient minimal solvers. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[39] Viktor Larsson and contributors. PoseLib - Minimal Solvers for Camera Pose Estimation, 2020. 6, 7, 8

[40] Viktor Larsson, Kalle Åström, and Magnus Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 5, 7

[41] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*. Springer, 2024. 3, 6, 7, 8

[42] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024. 2

[43] Hyon Lim, Sudipta N. Sinha, Michael F. Cohen, Matt Uyttendaele, and H. Jin Kim. Real-time Monocular Image-based 6-DoF Localization. *IJRR*, 34(4–5):476–492, 2015. 1

[44] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *International Conference on Computer Vision (ICCV)*, 2023. 6, 7, 8

[45] Sheng Liu, Xiaohan Nie, and Raffay Hamid. Depth-guided sparse structure-from-motion for movies and tv shows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2

[46] Stephan Liwicki and Christopher Zach. Scale exploiting minimal solvers for relative pose with calibrated cameras. In *BMVC*, 2017. 1, 2

[47] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2

[48] Gonzalo Martin Garcia, Karim Abou Zeid, Christian Schmidt, Daan de Geus, Alexander Hermans, and Bastian Leibe. Fine-tuning image-conditional diffusion models is easier than you think, 2024. 1, 2, 3

[49] Nathaniel Merrill, Patrick Geneva, Saimouli Katragadda, Chuchu Chen, and Guoquan Huang. Fast monocular visual-inertial initialization leveraging learned single-view depth. *Robotics: Science and Systems (RSS) 2023*, 2023. 2

[50] Oleg Naroditsky, Xun S Zhou, Jean Gallier, Stergios I Roumeliotis, and Kostas Daniilidis. Two efficient solutions for visual odometry using directional correspondence. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2012. 1

[51] David Nistér. An efficient solution to the five-point relative pose problem. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2004. 1, 2, 3, 4, 5, 6

[52] Mikael Persson and Klas Nordberg. Lambda twist: An accurate fast robust perspective three point (p3p) solver. In *Proceedings of the European conference on computer vision (ECCV)*, pages 318–332, 2018. 3, 4

[53] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10106–10116, 2024. 1, 2, 3, 6, 7, 8

[54] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *International Conference on Computer Vision (ICCV)*, 2021. 2, 3

[55] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 6

[56] 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*, 2020. 6

[57] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. 1

[58] 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 *Computer Vision and Pattern Recognition (CVPR)*, 2017. 5, 6, 7, 8

[59] Henrik Stewénius, David Nistér, Fredrik Kahl, and Frederik Schaffalitzky. A minimal solution for relative pose with unknown focal length. In *Computer Vision and Pattern Recognition (CVPR)*, 2005. 1

[60] Henrik Stewenius, Christopher Engels, and David Nistér. Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(4):284–294, 2006. 1, 2

[61] Linus Svärm, Olof Enqvist, Fredrik Kahl, and Magnus Oskarsson. City-scale localization for cameras with known vertical direction. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2016. 1

[62] Chris Sweeney, John Flynn, and Matthew Turk. Solving for relative pose with a partially known rotation is a quadratic eigenvalue problem. *International Conference on 3D Vision (3DV)*, 2014. 1

[63] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer vision and image understanding*, 2000. 3

[64] Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision, 2024. 1, 2, 3, 6, 7, 8

[65] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 2, 6

[66] Chen Yang, Sikuang Li, Jiemin Fang, Ruofan Liang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Gaussianobject: High-quality 3d object reconstruction from four views with gaussian splatting. *ACM Transactions on Graphics (TOG)*, 2024. 2

[67] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2, 3

[68] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024. 1, 2, 3, 6, 7, 8

[69] Yifan Yu, Shaohui Liu, Rémi Pautrat, Marc Pollefeys, and Viktor Larsson. Relative pose estimation through affine corrections of monocular depth priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16706–16716, 2025. 2, 3, 4, 5, 6, 7, 8

[70] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems*, 2022. 2

[71] Bernhard Zeisl, Torsten Sattler, and Marc Pollefeys. Camera pose voting for large-scale image-based localization. In *International Conference on Computer Vision (ICCV)*, 2015. 1

[72] Xinyue Zhang, Liangzu Peng, Wanting Xu, and Laurent Kneip. Accelerating globally optimal consensus maximization in geometric vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 4

[73] Yunwen Zhou, Abhishek Kar, Eric Turner, Adarsh Kowdle, Chao X Guo, Ryan C DuToit, and Konstantine Tsotsos. Learned monocular depth priors in visual-inertial initialization. In *European conference on computer vision*, 2022. 2