Relative Pose Estimation for Multi-Camera Systems from Point Correspondences with Scale Ratio

Banglei Guan
National University of Defense Technology
Changsha, Hunan, China
guanbanglei12@nudt.edu.cn

Ji Zhao*
Huazhong University of Science and Technology
Wuhan, Hubei, China
zhaoji84@gmail.com

ABSTRACT

The use of multi-camera systems is becoming more common in self-driving cars, micro aerial vehicles, or augmented reality headsets. In order to perform 3D geometric tasks, the accuracy and efficiency of relative pose estimation algorithms are very important for multi-camera systems, and are catching significant research attention these days. The point coordinates of point correspondences (PCs) obtained from feature matching strategies have been widely used for relative pose estimation. This paper exploits known scale ratios besides the point coordinates, which are also intrinsically provided by scale invariant feature detectors (*e.g.*, SIFT). Two-view geometry of scale ratio associated with the extracted features is derived for multi-camera systems. Thanks to the constraints provided by the scale ratio across two views, the number of PCs needed for relative pose estimation is reduced from 6 to 3. Requiring fewer PCs makes RANSAC-like randomized robust estimation significantly faster. For different point correspondence layouts, four minimal solvers are proposed for typical two-camera rigs. Extensive experiments demonstrate that our solvers have better accuracy than the state-of-the-art ones and outperform them in terms of processing time.

CCS CONCEPTS

• **Computing methodologies** → **Computer vision**.

KEYWORDS

Minimal solver, Generalized camera model, Multi-camera system, Relative pose estimation, Scale invariant feature

ACM Reference Format:
Banglei Guan and Ji Zhao. 2022. Relative Pose Estimation for Multi-Camera Systems from Point Correspondences with Scale Ratio. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3547788

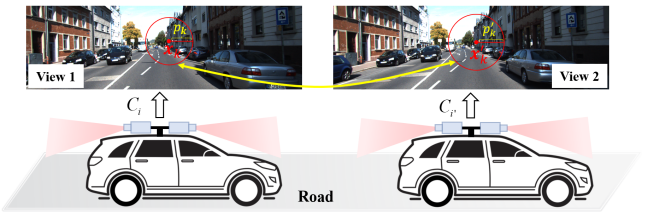


Figure 1: An point correspondence (PC) with scale ratio $(\mathbf{x}_k, \mathbf{x}'_k, r_k)$ in a multi-camera system relates two perspective cameras between consecutive views 1 and 2. $(\mathbf{x}_k, p_k)$ and $(\mathbf{x}'_k, p'_k)$ represent the point coordinates and feature scales in the i-th camera in the first view and the i'-th camera in the second view, respectively. PCs can be provided by scale invariant feature detectors. $r_k = p'_k/p_k$ represents the scale ratio, which is reciprocal to the ratio between depths of PC from two views.

1 INTRODUCTION

Estimating the relative poses from two views of a camera, or a multi-camera system is a fundamental component in modern 3D vision applications, such as robot localization and mapping, augmented reality, and autonomous driving [6, 22, 28, 29, 45, 48, 50, 51]. For example, augmented reality headsets usually have a multi-camera system for observing the environment. It is critical to accurately map the surrounding environments. Thus, there has always been much interest in developing accurate, efficient, and robust relative pose estimation algorithms [5, 12, 16, 18]. Motivated by the availability of multi-camera systems in various fields, including self-driving cars, micro aerial vehicles, or augmented reality headsets, this paper investigates the problem of relative pose estimation for multi-camera systems from point correspondences (PCs) with scale ratio, see Figure 1.

The single camera can be modeled by the perspective camera model [21]. The classical solvers for estimating the relative pose of a calibrated camera are the essential matrix algorithm with five PCs [25, 30, 40, 46] and the homography matrix algorithm with four PCs [21]. The multi-camera system is flexibly formed by fixing multiple perspective cameras to a single body. The combination of multiple cameras leads to that the multi-camera system can obtain more information about the environment. Compared to a single camera, a multi-camera system provides a large field-of-view, estimates in metric scale, and superior accuracy in the relative pose estimation. The main difference between a multi-camera system and a single camera is the absence of a single projection center [41]. To describe the light rays passed through the multi-camera system, Pless [41] proposed to express the light rays as Plücker lines and used the generalized essential constraint to describe the epipolar constraint of the Plücker lines. The relative pose of a multi-camera

*Corresponding author.

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547788

system can be estimated minimally from six PCs [47–49], linearly from seventeen PCs [31, 41] and iteratively from the optimization method [24]. The previously mentioned relative pose solvers estimate the pose parameters from a set of point coordinates, *e.g.*, coming from SIFT [34], SURF [7] or ORB [44] detectors.

It is generally admitted that the solver using a set of minimal correspondences performs more robustly, and requires fewer iterations when applying Random Sample Consensus (RANSAC) [13] as a robust estimator. Thus, a lot of methods exploit additional information besides the point coordinates to further reduce the number of minimal correspondences for both single camera [2, 4, 6, 8, 19, 20, 33, 43] and multi-camera system [1, 17, 18]. Recently, many works estimate the relative pose using affine correspondences (ACs), which contain more information about the underlying surface geometry than PCs [3]. The minimal number of necessary correspondences can be further reduced by replacing PCs with ACs [1, 6, 8, 17, 18, 43]. For example, Alyousefi and Ventura [1] estimated the relative pose of a multi-camera system from six ACs, which generalizes the linear solver from seventeen PCs proposed by Li *et al.* [31]. Guan *et al.* [17, 18] proposed several minimal solvers to compute the relative pose of a multi-camera system with additional motion constraints, such as planar motion, known vertical direction, and first-order approximation assumption. However, a major drawback of these methods in practice is that obtaining ACs, *e.g.*, via Affine SIFT [39], MODS [38] or Harris-Affine [35] detectors, is time-consuming. This severely restricts the applicability of these algorithms, especially for real-time applications [4, 11].

Inspired by the widely-used feature detectors such as SIFT [34], SURF [7] or ORB [44], which intrinsically provide a scale and rotation besides the point coordinates, a variety of solvers exploits the feature scales and rotations in the relative pose estimation. The significant advantage of these methods is that they can reduce the number of necessary correspondences. Typically, Liwicki *et al.* [33] estimated the relative pose of the single camera from four PCs with one known scale or three PCs with two known scales. Mills used four PCs with feature rotations to recover the relative pose of a single camera [36], and used a single PC with feature rotation to estimate purely rotational camera motion [37]. Barath *et al.* computed the fundamental matrix exploiting five PCs together with feature rotations [2], and estimated the homography from two PCs with their feature scales and rotations [4]. Ding *et al.* [11] solved the relative pose problem with known vertical direction from a single PC with its feature scale and rotation, while the points lie on the ground plane. However, the above mentioned works exploiting the scale and rotation components of features are only suitable for the single camera, rather than the multi-camera system.

In this paper, we aim at involving the feature scales in the relative pose estimation of a multi-camera system, which reduces the minimal number of necessary correspondences. To the best of our knowledge, this is the first paper to estimate the relative pose of a multi-camera system using PCs with scale ratios. The contributions of this paper are:

- We derive geometric observations relating scale ratio to the relative pose of a multi-camera system. A new general constraint holds for features which are obtained by the widely-used detectors, *e.g.*, SIFT, SURF, and ORB.
- We develop a series of minimal solvers for full DOF relative pose estimation problem of multi-camera systems, which cover common camera layouts and PC types of this problem. Our minimal solver generation exploits different rotation matrix parametrizations, which show that using quaternion parameterization results in smaller eliminate templates than using Cayley parameterization in our problem.
- Our minimal solvers are tested in a synthetic environment and on publicly available real-world datasets. The experimental results demonstrate that the proposed solvers are superior to the state-of-the-art methods in terms of accuracy and efficiency.

The remainder of the paper is structured as follows. Section 2 derives geometric constraints from PCs with scale ratio. Section 3 proposes our minimal solvers for the relative pose estimation of multi-camera systems. In Section 4, we evaluate the performance of proposed methods using both synthetic and real-world datasets. Finally, concluding remarks are given in Section 5.

2 GEOMETRIC CONSTRAINTS FROM PCS WITH SCALE RATIO

The intrinsic and extrinsic parameters of perspective cameras are known. The cameras satisfy the ideal pinhole camera model, and there are square pixels for the camera sensors. The minimal configurations for the relative pose estimation by using PCs with known scale ratios are explored in this section.

2.1 Two-View Geometry for A Single Camera

The k-th PC with known scale ratio is denoted as $(\mathbf{x}_k, \mathbf{x}'_k, r_k)$, where $\mathbf{x}_k$ and $\mathbf{x}'_k$ are the normalized homogeneous image coordinates in the first and the second views, respectively. $r_k = d_k/d'_k$ is the ratio between depths d_k and d'_k of PC $(\mathbf{x}_k, \mathbf{x}'_k)$ from two views. It has been proven that a PC with scale ratio yields two constraints, which are provided by the PC $(\mathbf{x}_k, \mathbf{x}'_k)$ and the scale ratio r_k, respectively [33].

The relative rotation and translation from the first view to the second view are represented as $(\mathbf{R}, \mathbf{t})$. The epipolar constraint derived from the PC $(\mathbf{x}_k, \mathbf{x}'_k)$ is written as [21]

$$(\mathbf{x}'_k)^T \mathbf{E} \mathbf{x}_k = 0, \tag{1}$$

where the essential matrix $\mathbf{E}$ is given as follows:

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}. \tag{2}$$

Since $r_k = d_k/d'_k$ is the scale ratio between depths d_k and d'_k of PC $(\mathbf{x}_k, \mathbf{x}'_k)$ from two views, it is observed in [33] that the direction of the translation $\mathbf{t}/\|\mathbf{t}\|$ is determined by rotation $\mathbf{R}$:

$$\mathbf{t} = d'_k \mathbf{x}'_k - d_k \mathbf{R} \mathbf{x}_k \;\Rightarrow\; (\mathbf{x}'_k - r_k \mathbf{R} \mathbf{x}_k) \times \mathbf{t} = \mathbf{0}. \tag{3}$$

Given k-th PC with scale ratio, a constraint for l-th PC can also be derived [33]:

$$\mathbf{x}'^T_l \mathbf{E} \mathbf{x}_l = 0 \;\Leftrightarrow\; (\mathbf{x}'_l \times \mathbf{x}'_k)^T \mathbf{R} \mathbf{x}_l - (\mathbf{x}'_l)^T \mathbf{R} (r_k \mathbf{x}_k \times \mathbf{x}_l) = 0, \tag{4}$$

Thus, given a set of PCs with scale ratios, we can choose each PC with a scale ratio respectively. Then other PCs are used to construct constraints as Eq. (4).

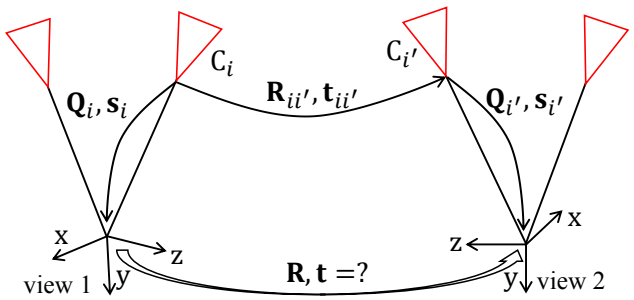


Figure 2: Relative pose estimation for a multi-camera system.

In summary, Eqs. (1) and (3) provide fundamental constraints provided by PC with known scale ratio. For non-degenerate cases, they provide two independent constraints only [33].

2.2 Two-View Geometry for A Multi-Camera System

A multi-camera system is made up of individual perspective cameras. The extrinsic parameters of the i-th camera expressed in a multi-camera reference frame are denoted as $\{\mathbf{Q}_i, \mathbf{s}_i\}$, where $\mathbf{Q}_i$ and $\mathbf{s}_i$ represent relative rotation and translation to the multi-camera reference frame, respectively. The relative rotation and translation from the first view to the second view of the multi-camera system are represented as $(\mathbf{R}, \mathbf{t})$.

As shown in Fig. 2, a PC with a scale ratio in a multi-camera system is seen by two different cameras across two views. The k-th PC with scale ratio is denoted as $(\mathbf{x}_k, \mathbf{x}'_k, i_k, i'_k, r_k)$. It means that a correspondence is captured by the i_k-th camera in the first view, and the normalized homogeneous image coordinate is $\mathbf{x}_k$. This correspondence is also captured by the i'_k-th camera in the second view, and the normalized homogeneous image coordinate is $\mathbf{x}'_k$. In the following text, the subscript k of camera indices i and i' is omitted in order to simplify the notation. The essential matrices for different correspondences are usually different in the multi-camera system. It is different from the two-view geometry for a single camera. Thus, the constraints derived from the k-th PC with scale ratio can be written as

$$(\mathbf{x}'_k)^T \mathbf{E}_k \mathbf{x}_k = 0 \tag{5a}$$

$$(\mathbf{x}'_k - r_k \mathbf{R}_{ii'} \mathbf{x}_k) \times \mathbf{t}_{ii'} = \mathbf{0} \tag{5b}$$

where

$$\mathbf{E}_k = [\mathbf{t}_{ii'}]_\times \mathbf{R}_{ii'}. \tag{6}$$

The relative rotation and translation from camera i in the first view to camera i' in the second view are represented as $(\mathbf{R}_{ii'}, \mathbf{t}_{ii'})$, which is determined by a composition of three transformations

$$\begin{bmatrix} \mathbf{R}_{ii'} & \mathbf{t}_{ii'} \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_{i'} & \mathbf{s}_{i'} \\ \mathbf{0} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{Q}_i & \mathbf{s}_i \\ \mathbf{0} & 1 \end{bmatrix} \tag{7}$$

$$= \begin{bmatrix} \mathbf{Q}_{i'}^T \mathbf{R} \mathbf{Q}_i & \mathbf{Q}_{i'}^T (\mathbf{R}\mathbf{s}_i + \mathbf{t} - \mathbf{s}_{i'}) \\ \mathbf{0} & 1 \end{bmatrix}. \tag{8}$$

By substituting Eq. (8) into Eq. (6), the essential matrix $\mathbf{E}_k$ is rewritten as

$$\mathbf{E}_k = \mathbf{Q}_{i'}^T (\mathbf{R}[\mathbf{s}_i]_\times + [\mathbf{t} - \mathbf{s}_{i'}]_\times \mathbf{R}) \mathbf{Q}_i. \tag{9}$$

By substituting Eqs. (8) and (9) into Eqs. (5a) and (5b), we obtain the fundamental two constraints for multi-camera systems

$$(\mathbf{Q}_{i'} \mathbf{x}'_k)^T (\mathbf{R}[\mathbf{s}_i]_\times + [\mathbf{t} - \mathbf{s}_{i'}]_\times \mathbf{R})(\mathbf{Q}_i \mathbf{x}_k) = 0, \tag{10}$$

and

$$\begin{aligned} &(\mathbf{x}'_k - r_k \mathbf{Q}_{i'}^T \mathbf{R} \mathbf{Q}_i \mathbf{x}_k) \times (\mathbf{Q}_{i'}^T (\mathbf{R}\mathbf{s}_i + \mathbf{t} - \mathbf{s}_{i'})) = \mathbf{0}, \\ \Rightarrow\ &\mathbf{x}'_k \times (\mathbf{Q}_{i'}^T (\mathbf{R}\mathbf{s}_i + \mathbf{t} - \mathbf{s}_{i'})) - r_k \mathbf{Q}_{i'}^T \mathbf{R} (\mathbf{Q}_i \mathbf{x}_k \times \mathbf{s}_i) \\ &- r_k \mathbf{Q}_{i'}^T (\mathbf{R} \mathbf{Q}_i \mathbf{x}_k \times (\mathbf{t} - \mathbf{s}_{i'})) = \mathbf{0}. \end{aligned} \tag{11}$$

It can be seen that Eqs. (10) and (11) are bilinear in the relative pose $(\mathbf{R}, \mathbf{t})$. If the k-th and l-th PCs with scale are captured by the same perspective cameras across two views, i.e., $i_k = i_l \triangleq i$ and $i'_k = i'_l \triangleq i'$, there are extra constraints which are similar to Eq. (4). When the scale ratio of k-th correspondence is used, we have

$$\begin{aligned} &(\mathbf{x}'_l \times \mathbf{x}'_k)^T \mathbf{R}_{ii'} \mathbf{x}_l - (\mathbf{x}'_l)^T \mathbf{R}_{ii'} (r_k \mathbf{x}_k \times \mathbf{x}_l) = 0, \\ \Rightarrow\ &(\mathbf{x}'_l \times \mathbf{x}'_k)^T (\mathbf{Q}_{i'}^T \mathbf{R} \mathbf{Q}_i) \mathbf{x}_l - (\mathbf{x}'_l)^T (\mathbf{Q}_{i'}^T \mathbf{R} \mathbf{Q}_i)(r_k \mathbf{x}_k \times \mathbf{x}_l) = 0. \end{aligned} \tag{12}$$

Moreover, when the scale ratio of l-th correspondence is used, we have a similar equation for extra constraints. For epipolar geometry between two views, it can be seen that the equations of the essential matrix are different in a single camera and a multi-camera system, which are expressed as Eq. (2) and Eq. (9), respectively. For a single camera, if the relative translation satisfies $\mathbf{t} = \mathbf{0}$ in the case of pure rotation, the essential matrix Eq. (2) reduces to zero. Thus, the epipolar constraint Eq. (1) is invalid [10]. However, for a multi-camera system, when the relative translation satisfies $\mathbf{t} = \mathbf{0}$, the essential matrix Eq. (9) would not reduce to zero, and the epipolar constraint Eq. (10) is still valid. Because the relative pose between two cameras in the multi-camera system is determined by a composition of three transformations, including the known extrinsic parameters of the cameras.

2.3 Minimal Configurations

For a single camera, there is 5DOF relative pose between two views. The relative pose of a single camera can be recovered by using three PCs with scale ratios [33], which provide six constraints. The excess constraint can be ignored during the solver generation procedure.

For multi-camera systems, there is 6DOF relative pose between two views. The relative pose estimation of a multi-camera system requires a minimal number of three PCs with scale ratios. This paper deals with typical two-camera rigs only, which is the most common scenario in practice. We exclude one symmetry between individual cameras and one symmetry between two views. There are four types of point correspondence with scale ratio, which are shown in Fig. 3. The correspondence type (i, i') means that the correspondence is captured by the i-th camera in the first view and the i'-th camera in the second view. For the 4 cases in Fig. 3, we name them as `3inter`, `3intra`, `2inter+1intra`, and `1inter+2intra` cases, respectively. The `inter` refers to inter-camera correspondences which are seen by different cameras over two consecutive views. They are suitable for two-camera rigs with extensive overlapping of views. The `intra` refers to intra-camera correspondences which are seen by the same

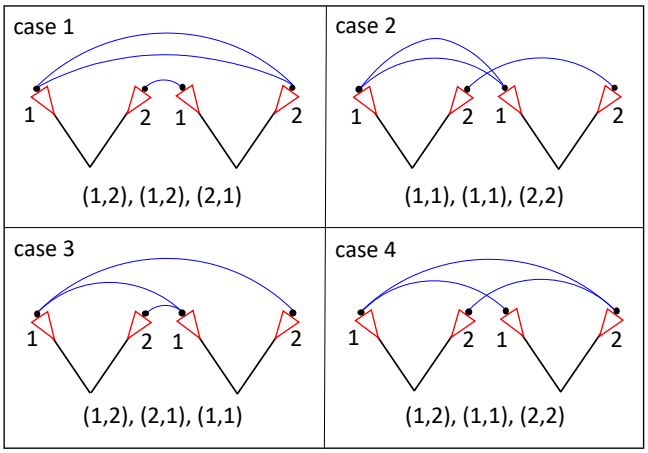


Figure 3: Four types of point correspondence with scale ratio. Case 1: `3inter`; Case 2: `3intra`, Case 3: `2inter+1intra`; Case 4: `1inter+2intra`. Red triangles represent perspective cameras in the multi-camera system; black dots represent correspondences; blue arcs represent the common observations that relate two perspective cameras between consecutive views. The correspondence type (i, i') means that the correspondence is captured by the i-th camera in the first view and the i'-th camera in the second view.

camera over two consecutive views. They are suitable for two-camera rigs with non-overlapping or small-overlapping of views.

Note that using all three correspondences from the same camera pair over two consecutive views is a degenerate case for the multi-camera solvers. Specifically, the rotation can be correctly recovered, while neither the translation direction nor the translation metric scale can be estimated. In this degenerate case, the relative pose estimation for a multi-camera system is simplified to the relative pose estimation for a single camera. The relative pose of the single camera can be estimated using three correspondences from the same camera [33]. It should be noted that the relative rotation $\mathbf{R}_{ii'}$ and the direction of the relative translation $\mathbf{t}_{ii'}$ can be recovered, but the metric scale of the relative translation $t_{ii'}$ cannot be recovered based on the single camera. Based on Eq. (7), with the known extrinsic parameters of the single camera $\mathbf{Q}_i, \mathbf{s}_i$, the relative rotation of the multi-camera system can be estimated, but the translation direction and the translation metric scale cannot be determined. Thus, to prevent this degenerate case, three correspondences in Fig. 3 are set to be split between the camera pairs over two consecutive views. Since the proposed solvers for the 4 cases need only three correspondences, they can be used efficiently for ego-motion estimation and outlier detection when integrating them into the RANSAC framework.

3 RELATIVE POSE ESTIMATION FROM PCS WITH SCALE RATIO

A series of minimal solvers for four cases in Fig. 3 is proposed in this section. We propose a complete solution to relative pose estimation for two-camera rigs, which uses the minimal number of three PCs with scale ratios.

3.1 Rotation Parameterization

The relative pose of a multi-camera system is first parameterized. There are several ways to parameterize the rotation matrix, such as Cayley, quaternions, and Euler angles. In the minimal problems, using the Cayley and quaternion parameterizations to formulate the problem has shown superiority [50]. The Cayley parameterization uses a homogeneous quaternion vector $[1, q_x, q_y, q_z]^T$, and the rotation matrix $\mathbf{R}$ can be written as

$$\mathbf{R}_{\text{cayl}} = \frac{1}{q_x^2+q_y^2+q_z^2+1} \begin{bmatrix} 1+q_x^2-q_y^2-q_z^2 & 2q_xq_y-2q_z & 2q_xq_z+2q_y \\ 2q_xq_y+2q_z & 1-q_x^2+q_y^2-q_z^2 & 2q_yq_z-2q_x \\ 2q_xq_z-2q_y & 2q_yq_z+2q_x & 1-q_x^2-q_y^2+q_z^2 \end{bmatrix}, \quad (13)$$

The Cayley parameterization introduces a degeneracy for 180° rotations, but this is a rare case for consecutive views [24, 50, 52]. The quaternion parameterization uses a normalized quaternion vector $\mathbf{q} = [q_w, q_x, q_y, q_z]^T$, and the rotation matrix $\mathbf{R}$ is given as

$$\mathbf{R}_{\text{quat}} = \begin{bmatrix} q_w^2+q_x^2-q_y^2-q_z^2 & 2q_xq_y-2q_wq_z & 2q_xq_z+2q_wq_y \\ 2q_xq_y+2q_wq_z & q_w^2-q_x^2+q_y^2-q_z^2 & 2q_yq_z-2q_wq_x \\ 2q_xq_z-2q_wq_y & 2q_yq_z+2q_wq_x & q_w^2-q_x^2-q_y^2+q_z^2 \end{bmatrix} \quad (14)$$

$$= \begin{bmatrix} 1-2(q_y^2+q_z^2) & 2q_xq_y-2q_wq_z & 2q_xq_z+2q_wq_y \\ 2q_xq_y+2q_wq_z & 1-2(q_x^2+q_z^2) & 2q_yq_z-2q_wq_x \\ 2q_xq_z-2q_wq_y & 2q_yq_z+2q_wq_x & 1-2(q_x^2+q_y^2) \end{bmatrix}, \quad (15)$$

where the normalized quaternion vector satisfying

$$q_w^2+q_x^2+q_y^2+q_z^2 = 1. \quad (16)$$

The quaternion parameterization does not have any degeneracy. The translation $\mathbf{t}$ is written as

$$\mathbf{t} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T. \quad (17)$$

3.2 Equation System and Solvers

In this paper, we investigate these two rotation parameterizations for the relative pose estimation of a multi-camera system. Take Cayley parameterization for an example. The solvers using Cayley parameterization are constructed in the following text. Our solver generation procedure can be applied to quaternion parameterization straightforwardly.

Based on Eqs. (10) and (11), the k-th correspondence provides four equations and there are six unknowns $\{q_x, q_y, q_z, t_x, t_y, t_z\}$. By multiplying a scale factor $q_x^2+q_y^2+q_z^2+1$, we obtain four polynomial equations. Since there are three correspondences in the minimal configurations, there are 12 cubic equations. In addition, for `3inter` and `3intra` cases, there are extra 2 quadratic equations as Eq. (12). A series of solvers for different cases can be obtained by the solver generator [26, 27]. We use `Macaulay 2` [15] to calculate Gröbner basis, and exploit both the Cayley and quaternion parameterizations.

Table 1 shows the minimal solvers for relative pose estimation of multi-camera systems using PCs with scale ratios. `#sym` indicates the number of symmetries, `#sol` indicates the number of solutions, and `template` indicates the size of elimination template. We have

Table 1: Relative pose estimation of multi-camera systems using PCs with scale ratios. cayl: Cayley, quat: quaternion, #sym: the number of symmetries, #sol: the number of solutions, and template: the size of elimination template.

configuration	3inter		3intra		2inter + 1intra		1inter + 2intra	
	cayl	quat	cayl	quat	cayl	quat	cayl	quat
#sym	0	1	0	1	0	1	0	1
#sol	4	8	4	8	8	16	8	16
template	203×207	134×138	203×207	134×138	444×452	331×339	444×452	331×339

the following observations. (1) For 3inter and 3intra cases, they have one-dimensional families of extraneous roots if the extra equations as Eq. (12) are not used. It verified the effectiveness of the extra constraints. (2) When quaternion is used, Eq. (15) results in smaller eliminate templates than that of Eq. (14). (3) The solvers using quaternion parameterization has multiple solutions with one symmetry [26], which corresponds to the sign ambiguity of the quaternion representation, *i.e.*, $\mathbf{R}_{\text{quat}}(\mathbf{q}) = \mathbf{R}_{\text{quat}}(-\mathbf{q})$. The size of the action matrix obtained by both Cayley parameterization and quaternion parametrization is the same. In addition, the number of rotation matrix solutions is also the same. (4) Using quaternion parameterization results in smaller eliminate templates than using Cayley parameterization. So the quaternion parameterization with Eq. (15) is recommended in the follow-up experiments.

Compared with six-point methods [47, 49], the proposed solvers have two advantages. First, it needs 3 correspondences instead of 6. When integrating it into a hypothesis-and-test framework, the proposed methods need significantly fewer iterations than six-point methods. Second, the proposed solvers have 8 or 16 solutions at most after excluding the solution symmetry. In contrast, six-point methods have 64 or 48 solutions at most [47, 49]. Hence the proposed solvers need less computation burden to reject false solutions.

4 EXPERIMENTS

In this section, extensive experiments on synthetic and real-world data are conducted to evaluate the performance of the proposed solvers. We perform the minimal solvers using quaternion parameterization. The proposed 3SIFT-Ours solvers are referred to as 3inter, 3intra, 2inter+1intra, and 1inter+2intra methods for the cases in Fig. 3, respectively. We compare the accuracy of the proposed solvers with the solvers 17PC-Li [31], 8PC-Kneip [24], 6PC-Stewénius [47], and 6AC-Ventura [1]. All the solvers are implemented in C++. The comparison solvers are provided by the OpenGV library [23] except that the code of 6AC-Ventura [1] is provided by its authors.

In the experiments, all the solvers are integrated within the RANSAC framework [13]. The correspondences that do not fulfill the two-view geometry for a multi-camera system would be identified as the outliers. We randomly select the correspondences for the methods. Thus, 17, 8, 6, 6, and 3 correspondences are selected randomly for the solvers 17PC-Li, 8PC-Kneip, 6PC-Stewénius, 6AC-Ventura, and 3SIFT-Ours, respectively, see supplementary material for details. We evaluate the solvers on the common configurations of correspondences in practice, which include inter-camera correspondences and intra-camera correspondences. The estimated error is measured on the relative pose, which produces the largest number of inliers within the RANSAC scheme. This also allows us to select the best candidate from multiple solutions by counting their inliers in a RANSAC-like procedure. An inlier threshold angle is set to 0.1° by following the definition in OpenGV [23]. The feasibility of our methods is demonstrated on KITTI dataset [14] and EuRoc MAV dataset [9].

The rotation error is computed as the angular difference between the ground truth rotation and the estimated rotation: $\varepsilon_{\mathbf{R}} = \arccos((\text{trace}(\mathbf{R}_{\text{gt}}\mathbf{R}^T) - 1)/2)$, where $\mathbf{R}_{\text{gt}}$ and $\mathbf{R}$ are the ground truth and estimated rotation matrices, respectively. Following the definition in [28, 42], the translation error is computed as $\varepsilon_{\mathbf{t}} = 2\left\|\mathbf{t}_{\text{gt}} - \mathbf{t}\right\| / (\left\|\mathbf{t}_{\text{gt}}\right\| + \|\mathbf{t}\|)$, where $\mathbf{t}_{\text{gt}}$ and $\mathbf{t}$ are the ground truth and estimated translations. The translation direction error is computed as $\varepsilon_{\mathbf{t},\text{dir}} = \arccos(\mathbf{t}_{\text{gt}}^T\mathbf{t}/(\|\mathbf{t}_{\text{gt}}\| \cdot \|\mathbf{t}\|))$.

4.1 Efficiency and Numerical Stability

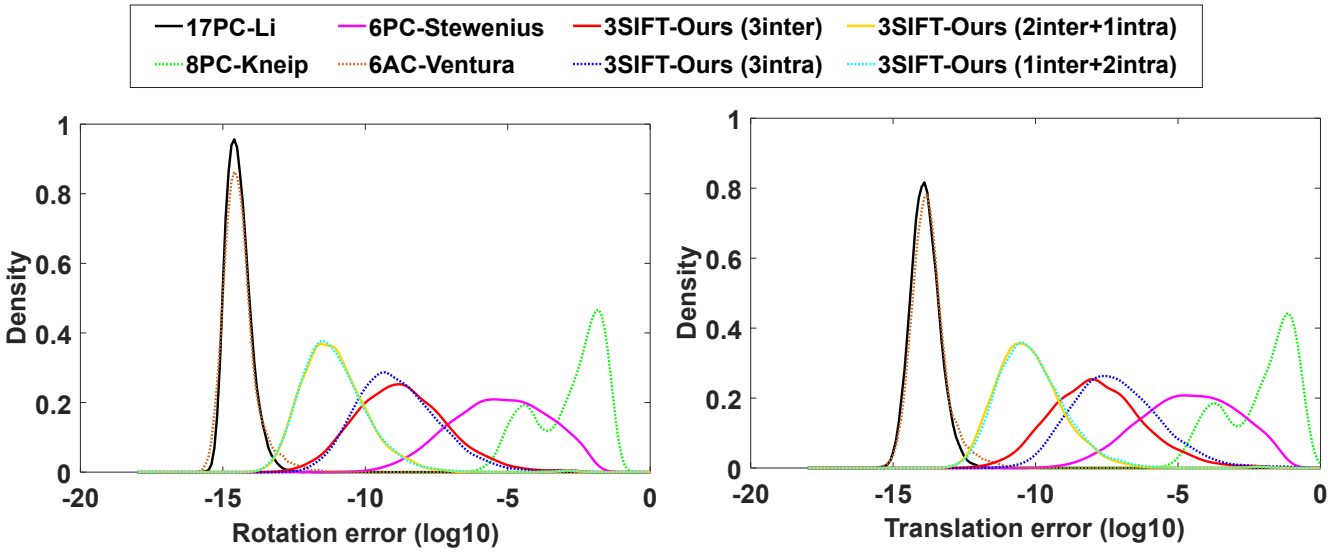
We evaluate the runtimes of the proposed solvers using an Intel(R) Core(TM) i7-7800X 3.50GHz. The average runtimes of the solvers over $10,000$ runs are shown in Table 2. The runtimes of the solvers 17PC-Li and 6AC-Ventura are low, because both solvers are linear solvers. As we will see later, since the proposed solvers require fewer correspondences, our solvers have better overall efficiency than the comparison solvers when integrating them into a RANSAC scheme.

The numerical stability of the solvers in the noise-free case is shown in Figure 4. We repeat the procedure $10,000$ times. The empirical probability density functions are plotted as the function of the $\log_{10}$ estimated errors. The solvers 17PC-Li [31] and 6AC-Ventura [1] have the best numerical stability, because both solvers are linear solvers and require the fewest calculations. Since 8PC-Kneip [24] uses the iterative optimization, it is susceptible to falling into local minima. Among the minimal solvers, all the proposed solvers 3inter, 3intra, 2inter+1intra, and 1inter+2intra have better numerical stability than the 6pt-Stewénius solver. It is interesting to see that the 2inter+1intra and 1inter+2intra have better numerical stability than the 3inter and 3intra. In addition, the 3intra has better numerical stability than the 3inter in rotation estimation.

In addition to efficiency and numerical stability, another important factor for a solver is the minimal number of required correspondences. Requiring fewer points makes RANSAC-like randomized robust estimation significantly faster. The iteration number N of RANSAC can be computed by $N = \log(1-p)/\log(1-(1-\epsilon)^s)$, where s is the minimal number of required correspondences, ϵ is the outlier ratio, and p is the success probability. For a success probability 99.9% and given a percentage of outliers $\epsilon = 50\%$, when the solvers require 17, 8, 6 and 3 correspondences, the iteration

Table 2: Runtime comparison of relative pose estimation solvers (unit: μs).

Methods	17PC-Li [31]	8PC-Kneip [24]	6PC-Stewénius [47]	6AC-Ventura [1]	**3inter**	**3intra**	**2inter+1intra**	**1inter+2intra**
Runtime	43.3	102.0	3275.4	38.1	231.2	232.7	2443.2	2441.7



(a) rotation error ε_{R} (b) translation error ε_{t}

Figure 4: Probability density functions over relative pose estimation errors in the noise-free case. The horizontal axis represents the $\log_{10}$ errors, and the vertical axis represents the density.

numbers of RANSAC are 905410, 1765, 439 and 52, respectively. As we will see later, the proposed solvers have better overall efficiency than the comparison solvers.

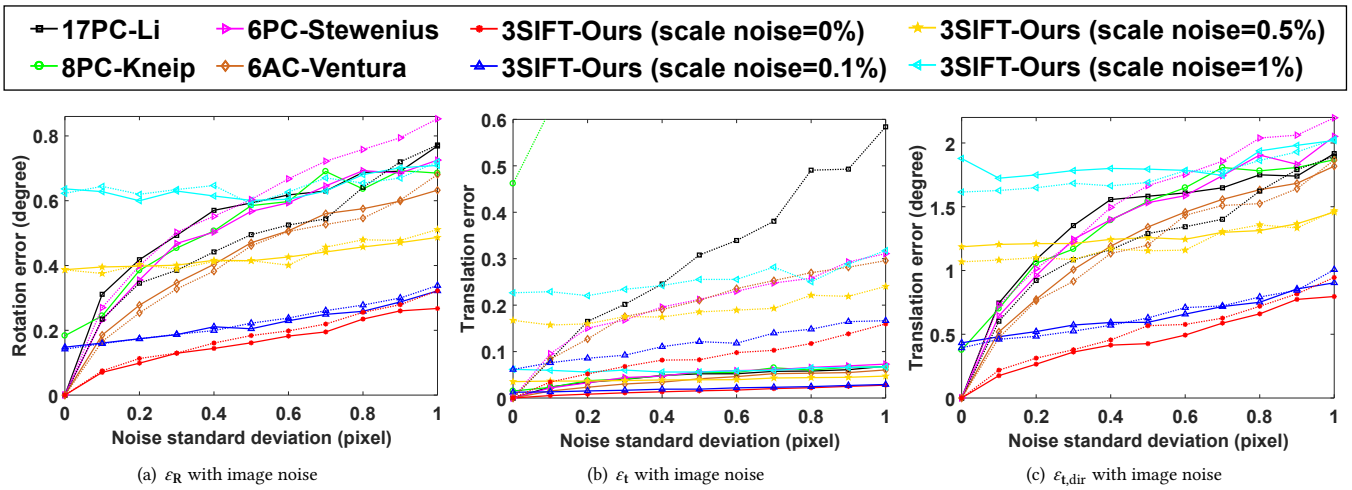
4.2 Experiments on Synthetic Data

We defined a simulated multi-camera system which can generate inter-camera correspondences and intra-camera correspondences simultaneously. The baseline length between the two simulated cameras is set to 1 meter. We define the multi-camera reference frame in the center of the camera rig and set the translation between two multi-camera reference frames to 3 meters. The focal lengths of the cameras are 400 pixels. The cameras' resolution is 640×480 pixels, and the principal points are set to the image center.

The synthetic scene is composed of a ground plane and 50 random planes. All 3D planes are randomly generated within the range of -5 to 5 meters (along axes X and Y), and 10 to 20 meters (Z-axis direction), which are expressed in the respective axis of the multi-camera reference frame. We choose 50 correspondences from the ground plane and one correspondence from each random plane randomly, thus, having a total of 100 correspondences. For each correspondence, the corresponding scale ratio is obtained by the ratio between depths of PC. For the `6AC-Ventura` solver, ACs are generated by following the procedure in [18], where the side length of the square is set as 30 pixels. A total of 1000 trials are carried out in the synthetic experiment. In each test, 100 correspondences are generated randomly. All the solvers are evaluated on both inter-camera correspondences and intra-camera correspondences.

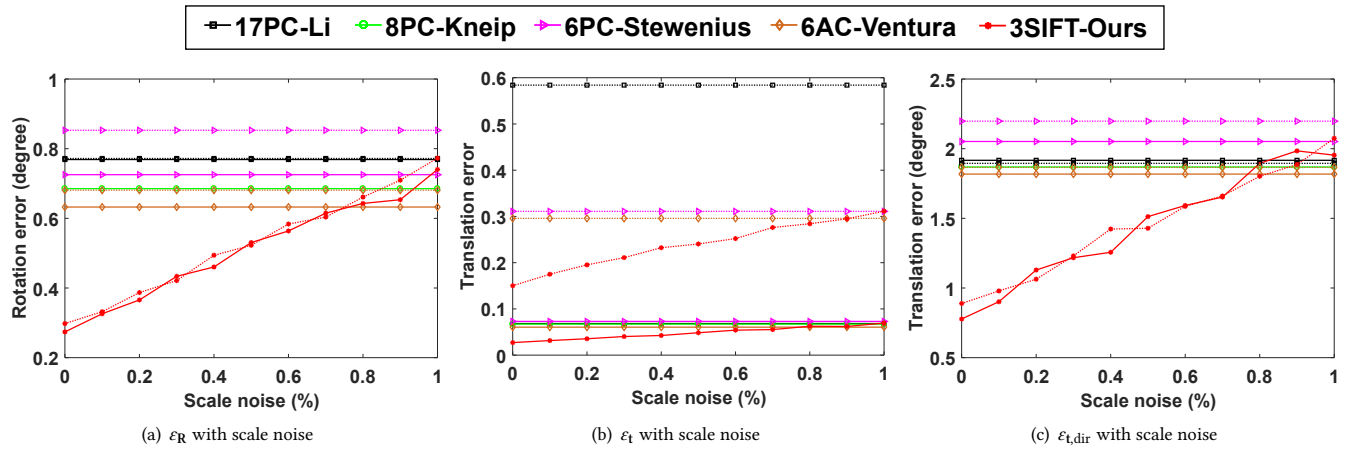
4.2.1 Accuracy with image noise. In this scenario, the magnitude of image noise is set to Gaussian noise with a standard deviation ranging from 0 to 1.0 pixel. The different levels of Gaussian noise with a standard deviation ranging from 0% to 1.0% are also added to the feature scales. The directions of the multi-camera system are set to forward, random, and sideways motions, respectively. Due to space limitations, we only show the results for random motion. Other results are in the supplementary material. Figure 5 shows the performance of the proposed solvers with increasing image noise under random motion. All the solvers are evaluated on both inter-camera correspondences and intra-camera correspondences. The corresponding estimation results are represented by solid lines and dotted lines, respectively. The `3SIFT-Ours` indicates `3inter` when using inter-camera correspondences, and indicates `3intra` when using intra-camera correspondences. In this figure, the display range is limited so that some curves with large errors are invisible or partially invisible.

We have the following observations. (1) The solvers using inter-camera correspondences generally have better performance than intra-camera correspondences, especially in recovering the metric scale of translation. (2) The performance of our `3SIFT-Ours` method is influenced by the noise magnitude of feature scales, which directly determines the accuracy of depth ratios from two views. Thus, our methods have better performance with lower



(a) ε_R with image noise (b) ε_t with image noise (c) $\varepsilon_{t,dir}$ with image noise

Figure 5: Rotation and translation error with increasing image noise under random motion. Solid line indicates using inter-camera correspondences, and dotted line indicates using intra-camera correspondences.



(a) ε_R with scale noise (b) ε_t with scale noise (c) $\varepsilon_{t,dir}$ with scale noise

Figure 6: Rotation and translation error with increasing scale noise under random motion. Solid line indicates using inter-camera correspondences, and dotted line indicates using intra-camera correspondences.

noise levels of the scales at the same magnitude of image noise. (3) When the noise level of the feature scales is less than 0.5%, the proposed `3SIFT-Ours` provides better results than the comparative methods with both inter-camera correspondences and intra-camera correspondences. (4) The `8PC-Kneip` performs well in the forward motion of the multi-camera systems, but it performs poorly in the random and sideways motion. The probable reason may be the iterative optimization which is susceptible to falling into local minima [48].

4.2.2 Accuracy with scale noise. In this scenario, the magnitude of scale noise is set to Gaussian noise with a standard deviation ranging from 0% to 1%. The image noise is set to 1.0 pixel standard deviation. Figure 6 shows the performance of the proposed solvers with increasing scale noise under random motion. The methods `17PC-Li`, `8PC-Kneip`, `6PC-Stewénius`, and `6AC-Ventura` are not influenced by the scale noise, because their calculation does not utilize the scale ratios.

We have the following observations. (1) The accuracy of our solvers decreases as the noise level of the feature scale increases. (2) The proposed `3SIFT-Ours` method outperforms the comparative methods in rotation estimation and translation direction estimation, even though the scale noise is around 0.7%. (3) Our solvers using inter-camera correspondences have better performance than intra-camera correspondences in recovering the metric scale of translation. (4) In comparison with using intra-camera correspondences, the proposed `3SIFT-Ours` method using inter-camera correspondences performs better in rotation estimation and translation direction estimation under sideways motion, and achieves comparable performance under forward and random motions.

Table 3: Rotation and translation error of the solvers on KITTI sequences (unit: degree).

Seq.	17PC-Li [31]		8PC-Kneip [24]		6PC-Stew. [47]		6AC-Vent. [1]		**3SIFT-Ours**	
	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$
00	0.147	2.537	0.148	2.496	0.259	4.938	0.153	2.708	**0.131**	**2.392**
01	0.178	4.407	0.182	**3.485**	0.303	7.557	**0.168**	3.879	0.185	3.803
02	0.142	1.988	0.147	2.094	0.228	3.474	0.150	2.441	**0.124**	**1.852**
03	**0.126**	2.762	0.139	2.833	0.327	6.324	0.144	3.069	0.136	**2.687**
04	0.113	1.733	0.123	1.829	0.269	3.770	0.127	1.939	**0.107**	**1.708**
05	0.132	2.663	0.130	2.461	0.225	4.404	0.136	2.315	**0.120**	**2.422**
06	0.139	2.146	0.151	2.145	0.203	3.258	0.128	2.187	**0.119**	**1.896**
07	0.131	3.085	0.172	3.259	0.264	6.831	0.151	2.998	**0.127**	**2.808**
08	0.133	2.705	0.135	2.762	0.228	4.914	0.128	2.957	**0.114**	**2.690**
09	0.144	2.022	0.138	**1.974**	0.212	3.246	0.146	2.159	**0.121**	1.982
10	0.142	2.398	0.141	2.393	0.249	4.155	0.221	2.469	**0.135**	**2.099**

Table 4: Runtime of RANSAC averaged over KITTI sequences combined with solvers (unit: *s*).

Methods	17PC-Li [31]	8PC-Kneip [24]	6PC-Stew. [47]	6AC-Vent. [1]	**3SIFT-Ours**
Mean time	3.157	0.648	4.723	0.384	**0.295**
Standard deviation	0.119	0.009	0.161	0.011	**0.008**

4.3 Experiments on Real-World Data

We evaluate the performance of the proposed solvers on the public datasets of two different kinds, namely the KITTI dataset [14] and the EuRoc MAV dataset [9]. These two datasets are collected on an autonomous driving and unmanned aerial vehicle environment, respectively, which are two popular modern robot applications. We compare the proposed solvers against state-of-the-art 6DOF relative pose estimation techniques.

4.3.1 Experiments on KITTI Dataset. We test the performance of our methods on the KITTI dataset [14], which consists of successive video frames from a forward-facing stereo camera. We ignore the overlap in their fields of view and treat it as a general multi-camera system. The sequences labeled from 00 to 10 that have ground truth are used for the evaluation. Therefore, the methods were tested on a total of 23000 image pairs. The PCs with scale ratio between consecutive frames in each camera are established by applying the SIFT [34]. In addition, the ACs are approximately generated by the PCs with scale ratio [4], which are used as input for the `6AC-Ventura` method [1]. The correspondences across the two cameras are not established to estimate the metric scale [24, 32]. All the solvers have been integrated into a RANSAC scheme to deal with outliers.

Table 3 shows the rotation and translation error of the proposed `3SIFT-Ours` for KITTI sequences. The median error is used to evaluate the performance. It is seen that the overall performance of the `3SIFT-Ours` outperforms the comparative methods in almost all cases. Moreover, to compare the advantage of computation efficiency, the RANSAC runtime averaged over all the KITTI sequences for the solvers is shown in Table 4. The reported runtimes include the relative pose estimation by RANSAC combined with a minimal solver. Since only three SIFTs are required for the proposed `3SIFT-Ours`, our method outperforms the comparative methods in terms of efficiency.

Table 5: Rotation and translation error of the solvers on EuRoC sequences (unit: degree).

Seq.	17PC-Li [31]		8PC-Kneip [24]		6PC-Stew. [47]		6AC-Vent. [1]		**3SIFT-Ours**	
	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$	$\varepsilon_{\mathbf{R}}$	$\varepsilon_{\mathbf{t},\mathrm{dir}}$
MH01	0.136	3.055	0.156	3.214	0.187	4.181	0.138	2.949	**0.115**	**2.764**
MH02	0.129	2.806	0.132	2.796	0.198	4.193	0.143	2.665	**0.125**	**2.416**
MH03	0.199	2.422	0.187	2.517	0.236	3.789	0.198	2.586	**0.167**	**2.312**
MH04	0.195	3.159	0.178	3.237	0.229	5.440	0.188	3.065	**0.171**	**2.723**
MH05	0.186	3.124	0.163	2.940	0.241	4.464	0.183	3.194	**0.152**	**2.818**

4.3.2 Experiments on EuRoC Dataset. To validate the proposed solver in an unmanned aerial vehicle environment, we further use the EuRoC MAV dataset [9] to evaluate the 6DOF relative pose estimation. The EuRoC MAV dataset is recorded using a stereo camera mounted on a micro aerial vehicle. We test the `3SIFT-Ours` solver on all the available five sequences, which are collected in a large industrial machine hall. Each sequence contains synchronized stereo images, accurate position, and IMU measurements. The observed scene is close to the cameras, and the image pairs present a significant change in viewpoint. The spatio-temporally aligned ground truth is provided from the nonlinear least-squares batch solution over the Leica position and IMU measurements. Since the industrial environment is unstructured and cluttered, it renders these sequences challenging to process. In order to prevent the movement of the image pair from being too small, the images for relative pose estimation are thinned out from the consecutive image sequences by an amount of one out of every four images. Besides, the image pairs with insufficient motion are cropped in this experiment. The PCs with scale ratio between consecutive frames in each camera have been established by applying the SIFT [34]. All the solvers are tested on about 3000 image pairs in total.

Table 5 shows the rotation and translation error of the proposed solvers for EuRoC sequences. It is shown that the `3SIFT-Ours` provides better results than the comparative methods `17PC-Li`, `8PC-Kneip`, `6PC-Stewénius` and `6AC-Ventura`. This experiment also demonstrates that our method is well suited for visual odometry in the unmanned aerial vehicle environment.

5 CONCLUSION

We proposed a complete solution and a series of solvers for relative pose estimation by exploiting the scale ratio of point correspondences. A minimum of three correspondences is used to estimate the 6DOF relative pose of a multi-camera system. Four minimal solvers using point correspondences with scale ratios are also proposed for two-camera rigs. Since the feature scales can be directly obtained when using the widely-used feature detectors, *e.g.* SIFT, our solvers can reduce the number of necessary correspondences. Based on a series of experiments on synthetic data and real-world image datasets, we demonstrate that our solvers can be used efficiently for ego-motion estimation and outperforms the state-of-the-art methods in both accuracy and efficiency.

ACKNOWLEDGMENTS

This work has been funded by the National Natural Science Foundation of China (Grant Nos. 11902349 and 11727804) and the Natural Science Foundation of Hunan Province (Grant No. 2020JJ5645).

REFERENCES

[1] Khaled Alyousefi and Jonathan Ventura. 2020. Multi-camera Motion Estimation with Affine Correspondences. In *International Conference on Image Analysis and Recognition*. 417–431.

[2] Daniel Barath. 2018. Five-point fundamental matrix estimation for uncalibrated cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*. 235–243.

[3] Daniel Barath and Levente Hajder. 2018. Efficient recovery of essential matrix from two affine correspondences. *IEEE Transactions on Image Processing* 27, 11 (2018), 5328–5337.

[4] Daniel Barath and Zuzana Kukelova. 2019. Homography from two orientation- and scale-covariant features. In *IEEE International Conference on Computer Vision*. 1091–1099.

[5] Daniel Barath, Michal Polic, Wolfgang FÃűrstner, Torsten Sattler, Tomas Pajdla, and Zuzana Kukelova. 2020. Making Affine Correspondences Work in Camera Geometry Computation. In *European Conference on Computer Vision*. 723–740.

[6] Daniel Barath, Tekla Toth, and Levente Hajder. 2017. A minimal solution for two-view focal-length estimation using two affine correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition*. 6003–6011.

[7] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. 2008. Speeded-up robust features (SURF). *Computer Vision and Image Understanding* 110, 3 (2008), 346–359.

[8] Jacob Bentolila and Joseph M Francos. 2014. Conic epipolar constraints from affine correspondences. *Computer Vision and Image Understanding* 122 (2014), 105–114.

[9] Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W Achtelik, and Roland Siegwart. 2016. The EuRoC micro aerial vehicle datasets. *The International Journal of Robotics Research* 35, 10 (2016), 1157–1163.

[10] Qi Cai, Yuanxin Wu, Lilian Zhang, and Peike Zhang. 2019. Equivalent constraints for two-view geometry: Pose solution/pure rotation identification and 3D reconstruction. *International Journal of Computer Vision* 127, 2 (2019), 163–180.

[11] Yaqing Ding, Daniel Barath, and Zuzana Kukelova. 2020. Homography-based Egomotion Estimation Using Gravity and SIFT Features. In *Asian Conference on Computer Vision*.

[12] Ivan Eichhardt and Daniel Barath. 2020. Relative Pose from Deep Learned Depth and a Single Affine Correspondence. In *European Conference on Computer Vision*. 627–644.

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

[14] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research* 32, 11 (2013), 1231–1237.

[15] Daniel R Grayson and Michael E Stillman. 2002. Macaulay 2, a software system for research in algebraic geometry. https://faculty.math.illinois.edu/Macaulay2/.

[16] Banglei Guan, Pascal Vasseur, Cédric Demonceaux, and Friedrich Fraundorfer. 2018. Visual odometry using a homography formulation with decoupled rotation and translation estimation using minimal solutions. In *IEEE International Conference on Robotics and Automation*. 2320–2327.

[17] Banglei Guan, Ji Zhao, Daniel Barath, and Friedrich Fraundorfer. 2021. Efficient Recovery of Multi-Camera Motion from Two Affine Correspondences. In *IEEE International Conference on Robotics and Automation*. 1305–1311.

[18] Banglei Guan, Ji Zhao, Daniel Barath, and Friedrich Fraundorfer. 2021. Minimal Cases for Computing the Generalized Relative Pose Using Affine Correspondences. In *IEEE International Conference on Computer Vision*. 6068–6077.

[19] Banglei Guan, Ji Zhao, Zhang Li, Fang Sun, and Friedrich Fraundorfer. 2020. Minimal Solutions for Relative Pose With a Single Affine Correspondence. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1929–1938.

[20] Banglei Guan, Ji Zhao, Zhang Li, Fang Sun, and Friedrich Fraundorfer. 2021. Relative Pose Estimation With a Single Affine Correspondence. *IEEE Transactions on Cybernetics* (2021), 1–12. https://doi.org/10.1109/TCYB.2021.3069806

[21] Richard Hartley and Andrew Zisserman. 2003. *Multiple view geometry in computer vision*. Cambridge University Press.

[22] Tiejun Huang, Yajing Zheng, Zhaofei Yu, Rui Chen, Yuan Li, Ruiqin Xiong, Lei Ma, Junwei Zhao, Siwei Dong, Lin Zhu, et al. 2022. 1000× Faster Camera and Machine Vision with Ordinary Devices. *Engineering* (2022).

[23] Laurent Kneip and Paul Furgale. 2014. OpenGV: A unified and generalized approach to real-time calibrated geometric vision. In *IEEE International Conference on Robotics and Automation*. 12034–12043.

[24] Laurent Kneip and Hongdong Li. 2014. Efficient computation of relative pose for multi-camera systems. In *IEEE Conference on Computer Vision and Pattern Recognition*. 446–453.

[25] Laurent Kneip, Roland Siegwart, and Marc Pollefeys. 2012. Finding the exact rotation between two images independently of the translation. In *European Conference on Computer Vision*. Springer, 696–709.

[26] Viktor Larsson and Kalle Aström. 2016. Uncovering symmetries in polynomial systems. In *European Conference on Computer Vision*. Springer, 252–267.

[27] Viktor Larsson, Kalle Aström, and Magnus Oskarsson. 2017. Efficient Solvers for Minimal Problems by Syzygy-based Reduction. In *IEEE Conference on Computer Vision and Pattern Recognition*. 820–828.

[28] Gim Hee Lee, Marc Pollefeys, and Friedrich Fraundorfer. 2014. Relative pose estimation for a multi-camera system with known vertical direction. In *IEEE Conference on Computer Vision and Pattern Recognition*. 540–547.

[29] Bo Li, Evgeniy Martyushev, and Gim Hee Lee. 2020. Relative Pose Estimation of Calibrated Cameras with Known SE(3) Invariants. In *European Conference on Computer Vision*. 215–231.

[30] Hongdong Li and Richard Hartley. 2006. Five-Point Motion Estimation Made Easy. In *International Conference on Pattern Recognition*. 630–633.

[31] Hongdong Li, Richard Hartley, and Jae-hak Kim. 2008. A linear approach to motion estimation using generalized camera models. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1–8.

[32] Liu Liu, Hongdong Li, Yuchao Dai, and Quan Pan. 2017. Robust and efficient relative pose with a multi-camera system for autonomous driving in highly dynamic environments. *IEEE Transactions on Intelligent Transportation Systems* 19, 8 (2017), 2432–2444.

[33] Stephan Liwicki and Christopher Zach. 2017. Scale Exploiting Minimal Solvers for Relative Pose with Calibrated Cameras. In *British Machine Vision Conference*.

[34] David G Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60, 2 (2004), 91–110.

[35] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and L Van Gool. 2005. A comparison of affine region detectors. *International journal of computer vision* 65, 1 (2005), 43–72.

[36] Steven Mills. 2018. Four-and seven-point relative camera pose from oriented features. In *2018 International Conference on 3D Vision (3DV)*. IEEE, 218–227.

[37] Steven Mills. 2021. Relative Camera Rotation from a Single Oriented Correspondence. In *International Conference on Image and Vision Computing New Zealand*. IEEE, 1–6.

[38] Dmytro Mishkin, Jiri Matas, and Michal Perdoch. 2015. MODS: Fast and robust method for two-view matching. *Computer Vision and Image Understanding* 141 (2015), 81–93.

[39] Jean-Michel Morel and Guoshen Yu. 2009. ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences* 2, 2 (2009), 438–469.

[40] David Nistér. 2004. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 6 (2004), 756–777.

[41] Robert Pless. 2003. Using many cameras as one. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1–7.

[42] Long Quan and Zhongdan Lan. 1999. Linear n-point camera pose determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21, 8 (1999), 774–780.

[43] Carolina Raposo and Joao P Barreto. 2016. Theory and practice of structure-from-motion using affine correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition*. 5470–5478.

[44] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. 2011. ORB: An efficient alternative to SIFT or SURF. In *International Conference on Computer Vision*. 2564–2571.

[45] Davide Scaramuzza and Friedrich Fraundorfer. 2011. Visual odometry: The first 30 years and fundamentals. *IEEE Robotics & Automation Magazine* 18, 4 (2011), 80–92.

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

[47] Henrik Stewénius, Magnus Oskarsson, Kalle Aström, and David Nistér. 2005. Solutions to minimal generalized relative pose problems. In *Workshop on Omnidirectional Vision in conjunction with ICCV*. 1–8.

[48] Jonathan Ventura, Clemens Arth, and Vincent Lepetit. 2015. An efficient minimal solution for multi-camera motion. In *IEEE International Conference on Computer Vision*. 747–755.

[49] Ji Zhao and Banglei Guan. 2021. On Relative Pose Recovery for Multi-Camera Systems. *arXiv:2102.11996* (2021).

[50] Ji Zhao, Laurent Kneip, Yijia He, and Jiayi Ma. 2020. Minimal case relative pose computation using ray-point-ray features. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 5 (2020), 1176–1190.

[51] Ji Zhao, Wanting Xu, and Laurent Kneip. 2020. A Certifiably Globally Optimal Solution to Generalized Essential Matrix Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 12034–12043.

[52] Enliang Zheng and Changchang Wu. 2015. Structure From Motion Using Structure-Less Resection. In *IEEE International Conference on Computer Vision*. 2075–2083.