A Minimal Solution to the Generalized Pose-and-Scale Problem

Jonathan Ventura
Institute for Computer Graphics and Vision
Graz University of Technology, Austria
ventura@icg.tugraz.at

Clemens Arth
Institute for Computer Graphics and Vision
Graz University of Technology, Austria
arth@icg.tugraz.at

Gerhard Reitmayr
Qualcomm Austria Research Center
Vienna, Austria
gerhardr@qti.qualcomm.com

Dieter Schmalstieg
Institute for Computer Graphics and Vision
Graz University of Technology, Austria
schmalstieg@tugraz.at

Abstract

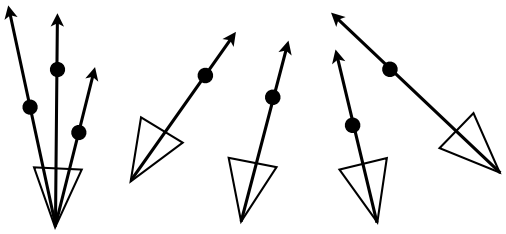
We propose a novel solution to the generalized camera pose problem which includes the internal scale of the generalized camera as an unknown parameter. This further generalization of the well-known absolute camera pose problem has applications in multi-frame loop closure. While a well-calibrated camera rig has a fixed and known scale, camera trajectories produced by monocular motion estimation necessarily lack a scale estimate. Thus, when performing loop closure in monocular visual odometry, or registering separate structure-from-motion reconstructions, we must estimate a seven degree-of-freedom similarity transform from corresponding observations.

Existing approaches solve this problem, in specialized configurations, by aligning 3D triangulated points or individual camera pose estimates. Our approach handles general configurations of rays and points and directly estimates the full similarity transformation from the 2D-3D correspondences. Four correspondences are needed in the minimal case, which has eight possible solutions. The minimal solver can be used in a hypothesize-and-test architecture for robust transformation estimation. Our solver also produces a least-squares estimate in the overdetermined case.

The approach is evaluated experimentally on synthetic and real datasets, and is shown to produce higher accuracy solutions to multi-frame loop closure than existing approaches.

1. Introduction

A well-known classical problem in photogrammetry is the determination of the absolute pose of a calibrated camera given three imaged point observations, for which several solutions exist (*c.f.* [16]). In the generalized camera pose problem, the imaging rays do not necessarily meet at a common optical center [4, 25]. In this paper, we propose a new absolute pose problem which is a further generalization, including the internal scale of the generalized camera as an unknown parameter. As illustrated in Figure 1, multiple cameras, or the images from a single, moving camera, can be modeled as a single generalized camera, described by its imaging rays. In the generalized pose-and-scale problem, we wish to determine the position and orientation of the set of cameras, as well as the scale of the translation between them, with respect to a set of observed anchor points.



(a) P3P (b) gP+s

Figure 1: (a) From three or more point observations we can recover the pose of a pinhole camera, where all image rays intersect at a common optical center. (b) In this paper we consider the problem of solving for both the pose and scale of a generalized camera, where the rays do not necessarily meet at a common optical center, from four or more point observations. The generalized camera represents multiple cameras, or, multiple images from a single, moving camera, as one.

This problem arises in monocular camera motion estimation and scene reconstruction, or structure-from-motion (SfM). It is impossible to determine scale from camera images alone, without some external piece of information, such as a measured distance between cameras or a mea-

sured dimension of an observed object. Thus, when registering two SfM results, the relative scale must be determined along with the translation and rotation. For example, when integrating loop closures in visual odometry, when joining two independent SfM reconstructions together, or when registering a single SfM reconstruction with known anchor points, the generalized pose+scale problem must be solved. Essentially, we must find the similarity transform which relates the two coordinate systems, given image observations in one reconstruction of 3D points in the other reconstruction.

Existing solutions to loop closure and SfM alignment have typically used some combination of single-image pose determination, point triangulation, and scale estimation. One common approach is to match triangulated 3D points between the two reconstructions, and then use absolute orientation [13, 34] to determine the seven degree-of-freedom registration [6]. A second approach is to localize two images separately; one localized image determines the rotation and translation, and the distance to the other image determines the scale [36]. A third approach essentially combines the first two: a single image is localized, and the distance to points triangulated using a second camera determines the scale [30]. These approaches represent solutions to specific instances of the more general pose+scale problem: namely, having three points or two cameras.

The solution proposed in this work handles the aforementioned specific cases, as well as all other configurations: three or four unique points, and two, three, or four cameras, without requiring multiple observations of points, or a specific number of correspondences in any one camera. This makes our algorithm simple to apply in a robust sampling procedure such as RANSAC [11] or PROSAC [5]. Our solution also provides a least squares answer in the overdetermined case. Our experimental results show that our solution has better accuracy than other special-case alignment methods.

In the following, Section 2 discusses related work in the area of minimal solvers and the corresponding application scenario. Section 3 gives a formal problem statement followed by a description of the proposed solution in Section 4. Experimental results on synthetic and real data are described in Section 5. Concluding remarks are given in Section 6.

2. Related work

Recently, there has been a great deal of work on minimal solvers for absolute and relative camera pose problems [1, 2, 3, 17, 18, 19, 21, 22, 24, 29]. A fast and numerically stable solution which requires the minimal number of correspondences is useful because it can then be applied in a random sampling framework to robustly find the solution most consistent with the contaminated input set [11].

The problem we solve belongs to the group of Non-Perspective-n-Point (NPnP) problems. Closest related to our work are solutions to the NP3P problem, where minimal solutions were proposed by Chen and Chang [4] and Nistér [25]. Chen and Chang developed an iterative solver for the NPnP problem with the NP3P problem as a special case. Nistér developed a near closed-form solution for the minimal case of three point-ray correspondences; the problem is reduced to the solution of an octic polynomial, which, although it cannot be solved in closed-form, can be solved efficiently by transferring the problem to an eigenvalue problem or using other root-finding methods [3]. Other solutions to the NPnP problem were proposed by Schweighofer and Pinz [28] and by Tariq and Dellaert [32]. Kneip *et al.* [15] describe an approach which is a modification to the ePnP algorithm originally proposed by Lepetit *et al.* [20]. The so-called gPnP algorithm gives an $O(n)$ solution to the overdetermined NPnP problem, employing Gröbner basis computations to resolve the different cases arising from noisy correspondence samples. We known of no previous solution to the generalized pose and scale problem, which we call gP+s, or alternatively, NP4P+s.

A potential application of the proposed algorithm is in the field of online visual odometry and SLAM, where different strategies for loop closure are found [7, 9, 10, 14, 33]. These solutions usually contain variants, derivatives or combinations of either the absolute orientation algorithm used to align particles or landmarks, or use PnP algorithms applied on single or multiple images. This is usually followed by backpropagation of correction terms into previously seen structure and previous pose estimates, setting up a local or global error minimization problem using Bundle Adjustment (BA). Essentially, the solution we propose in this work can be considered a replacement for the similarity transformation estimation step, whenever registering a set of cameras with a point cloud. Therefore, it is suitable for inclusion into these kind of approaches.

3. Problem statement

Notation: We use a bold capital letter $\mathbf{M}$ for a matrix, a bold lower-case letter $\mathbf{v}$ for a vector, and an italic lower-case letter s for a scalar.

Our goal is to determine the pose and internal scale, with respect to known anchor points, of a generalized camera which is described by n rays. We describe a ray of the generalized camera with a starting point $\mathbf{p}_i$ and a direction $\mathbf{d}_i$. The corresponding anchor point is denoted $\mathbf{q}_i$. The vectors $\mathbf{p}_i$, $\mathbf{d}_i$, and $\mathbf{q}_i$ each have size 3×1. We want to find a rotation $\mathbf{R}$, translation $\mathbf{t}$ and scale s so that the rays coincide with the points:

$$\mathbf{R}\mathbf{q}_i + \mathbf{t} = s\mathbf{p}_i + \alpha_i \mathbf{d}_i \tag{1}$$

where α_i is an unknown scalar which stretches the ray

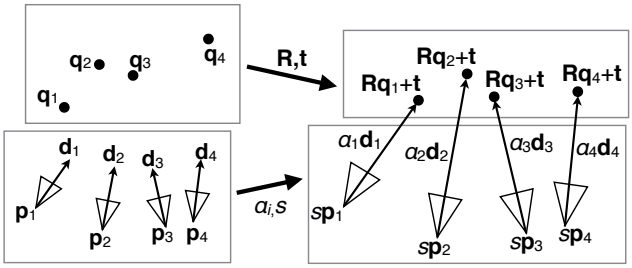


Figure 2: The input to the problem is at least four input points (top left) and four corresponding input rays (bottom left). The solver computes all possible similarity transforms which align them. Each transformation is composed of a rotation and translation applied to the points (top right) and a scale applied to the ray origins (bottom right). The ray stretch factors α_i are also illustrated here but do not need to be computed.

to meet the anchor point when the generalized camera is aligned. See Figure 2 for an illustration of the problem.

4. Solution procedure

We first re-arrange equation 1 and use the fact that the cross product of coincident vectors is zero to eliminate the unknown scalar α_i:

$$\mathbf{d}_i \times (\mathbf{R}\mathbf{q}_i + \mathbf{t} - s\mathbf{p}_i) = \alpha_i(\mathbf{d}_i \times \mathbf{d}_i) = \mathbf{0}. \quad (2)$$

This gives three equations, only two of which are linearly independent. We have seven degrees of freedom: three for rotation, three for translation and one for scale. Thus in the minimal case we need four point-ray correspondences for the system to have a finite number of solutions.

Collecting the unknown parameters into a vector $\mathbf{x}$, the two linearly independent equations from Equation 2 are:

$$\mathbf{a}_{i1}\mathbf{x} = 0 \quad (3)$$
$$\mathbf{a}_{i2}\mathbf{x} = 0 \quad (4)$$

where

$$\mathbf{a}_{i1} \equiv [\mathbf{0}^\top \; -d_{i3}\mathbf{q}_i^\top \; d_{i2}\mathbf{q}_i^\top \; 0 \; -d_{i3} \; d_{i2} \; (d_{i3}p_{i2} - d_{i2}p_{i3})]$$
$$\mathbf{a}_{i2} \equiv [d_{i3}\mathbf{q}_i^\top \; \mathbf{0}^\top \; -d_{i1}\mathbf{q}_i^\top \; d_{i3} \; 0 \; -d_{i1} \; (d_{i1}p_{i3} - d_{i3}p_{i1})]$$
$$\mathbf{x} \equiv [\, \mathbf{r}_1 \; \mathbf{r}_2 \; \mathbf{r}_3 \; t_1 \; t_2 \; t_3 \; s \,]^\top. \quad (5)$$

Here $\mathbf{r}_j$ denotes the j-th row of matrix $\mathbf{R}$ and t_j denotes the j-th element of $\mathbf{t}$.

By stacking equations 3 and 4 from all correspondences into a single matrix $\mathbf{A}$, we arrive at a linear system of $2n$ equations:

$$\mathbf{A}\mathbf{x} = \mathbf{0} \quad (6)$$

In the minimal case, where $n = 4$, we could remove one equation to have the minimum of seven equations. However, we found that this can cause the system to be rank-deficient, so we always solve the complete system of $2n$ equations.

Directly solving the linear system $\mathbf{A}\mathbf{x} = \mathbf{0}$ does not guarantee that the matrix $\mathbf{R}$ will have the properties of a rotation matrix, *i.e.* that $\mathbf{R}$ is orthogonal and $\det(\mathbf{R}) = 1$. Instead, we extract the six vectors $\mathbf{b}_1, \ldots, \mathbf{b}_6$ which span the right nullspace of matrix $\mathbf{A}$. This can be achieved using the singular value decomposition or the QR decomposition [27]. Any solution for $\mathbf{x}$ is then a linear combination of the six null basis vectors:

$$\mathbf{x} = \beta_1\mathbf{b}_1 + \ldots + \beta_6\mathbf{b}_6 \quad (7)$$

for some scalars $\beta_1, \ldots, \beta_6$. Similar to Nistér's solution to the five-point relative pose problem [24], this procedure can be extended to use more than four correspondences. In the overdetermined case, we extract the six singular vectors which correspond to the six smallest singular values.

We can remove one variable by fixing $\beta_6 = 1$. The following ten quadratic constraints ensure that the matrix $\mathbf{R}$ is orthogonal, up to scale:

$$||\mathbf{r}_1||^2 - ||\mathbf{r}_2||^2 = 0 \quad (8)$$
$$||\mathbf{r}_1||^2 - ||\mathbf{r}_3||^2 = 0 \quad (9)$$
$$||\mathbf{c}_1||^2 - ||\mathbf{r}_2||^2 = 0 \quad (10)$$
$$||\mathbf{c}_1||^2 - ||\mathbf{c}_3||^2 = 0 \quad (11)$$
$$\mathbf{r}_1 \cdot \mathbf{r}_2 = 0 \quad (12)$$
$$\mathbf{r}_1 \cdot \mathbf{r}_3 = 0 \quad (13)$$
$$\mathbf{r}_2 \cdot \mathbf{r}_3 = 0 \quad (14)$$
$$\mathbf{c}_1 \cdot \mathbf{c}_2 = 0 \quad (15)$$
$$\mathbf{c}_1 \cdot \mathbf{c}_3 = 0 \quad (16)$$
$$\mathbf{c}_2 \cdot \mathbf{c}_3 = 0 \quad (17)$$

where $\mathbf{c}_j$ denotes the j-th column of matrix $\mathbf{R}$.

By inserting Equation 7 into these constraints, we arrive at ten quadratic equations in twenty-one monomials with the variables $\beta_1, \ldots, \beta_5$. After extracting the roots of this system of equations (see next section), for each solution we divide $\mathbf{x}$ by $||\mathbf{r}_1||$ and negate $\mathbf{x}$ if necessary to make $\det(\mathbf{R}) = 1$.

4.1. Gröbner basis solution

We used the automatic tool of Kukelova, Bujnak and Pajdla [17] to find a reduced Gröbner basis using the *grevlex* monomial ordering [8] for the system of ten polyonomial equations described above. The system of polynomials has eight solutions in general. The produced solver performs row reduction via LU decomposition on a coefficient matrix $\mathbf{M}$ of size 48×56, and from the row-reduced matrix extracts coefficients to produce an 8×8 action matrix. The eigenvectors of this action matrix give the eight solutions for $\mathbf{x}$, from which we only keep the real-valued solutions. Furthermore, in most practical cases, we can reject any solution with negative scale. Finally, a unique solution can be

chosen by selecting the solution with the minimum angular error over all input rays.

The average time taken for each step of the solver, over 10^4 trials, is reported in Table 1.

SVD	LU	Eig.	Total
26.33	47.94	20.02	102.74

Table 1: Average computation time of the major steps of the minimal solver, in μs. The test was run on a 2.5 GHz Intel i7 machine, using a C++ implementation with the Eigen linear algebra library.

We expect that the speed of the method could be improved with the alternate root-finding methods suggested by Bujnak, Kukelova and Pajdla [3].

4.2. Degenerate cases

As in all absolute camera pose problems, if the anchor points are all collinear, no unique solution can be found, because there is a rotational ambiguity about the line connecting the points. This means that we need at least three unique and non-collinear anchor points to have a finite number of solutions. A second degenerate configuration arises when the rays $\mathbf{d}_i$ are all parallel. These degenerate cases cause the rank of matrix $\mathbf{A}$ from Equation 6 to drop, and can be detected numerically by checking whether $\mathbf{A}$ is ill-conditioned. The condition number $\kappa(\mathbf{A})$ is the ratio of the largest and smallest singular values of $\mathbf{A}$. If this ratio is very large, then the matrix is ill-conditioned, and thus the solution will be numerically unstable.

A third degenerate configuration arises when all the camera rays share the same optical center, meaning that the generalized camera could be modeled as a pinhole camera. While Nistér's solution [26] to the generalized camera pose problem, without scale, can seamlessly handle the pinhole camera case, for our problem it is not possible. This is because without any baseline between the rays, the scale parameter s is unconstrained. In practical scenarios, the situation might arise that the distance to the visible anchor points is large relative to the baseline between camera rays. In this situation, the scale of the generalized camera is close to zero, and thus the camera configuration is close to having a single optical center with respect to the anchor points.

To show the effect of having a common optical center on our solution, we first define $\mathbf{A}_{\mathbf{t},s}$ as the last four columns of $\mathbf{A}$, corresponding to the translation and scale components. By including all three equations from Equation 2, we see that $\mathbf{A}_{\mathbf{t},s}$ can be expressed simply as

$$\mathbf{A}_{\mathbf{t},s} \equiv [\ [\mathbf{d}_i]_\times\ [\mathbf{d}_i]_\times \mathbf{p}_i\ ] \tag{18}$$

where $[\mathbf{d}_i]_\times$ is a skew-symmetric matrix such that $[\mathbf{d}_i]_\times \mathbf{v} = \mathbf{d}_i \times \mathbf{v}$ for any vector $\mathbf{v}$ [12]. If the ray origins $\mathbf{p}_i$ are equal, *i.e.*, $\mathbf{p}_i = \mathbf{p}\ \forall i$, then the rank of $\mathbf{A}_{\mathbf{t},s}$ will drop, because then the last column is a linear combination of the first three. More generally, we have the same situation if each $\mathbf{p}_i = \mathbf{p} + \lambda_i \mathbf{d}_i$ for some unknown center point $\mathbf{p}$ and scalars $\lambda_1, \ldots, \lambda_n$. This is because, plugging into Equation 18, we can see that $[\mathbf{d}_i]_\times(\mathbf{p} + \lambda_i \mathbf{d}_i) = [\mathbf{d}_i]_\times \mathbf{p}$.

From this analysis, we can see that if $\mathbf{A}_{\mathbf{t},s}$ is ill-conditioned, then the problem is better solved by ignoring the $\mathbf{p}_i$ and assuming that all rays emanate from a single point. In this case, we can use an existing P3P [16] or PnP [20] solver, and assume that $s = 1$. In Section 5.2, we investigate numerically the relationship between the generalized camera model and the pinhole camera model as the baseline between the camera rays changes.

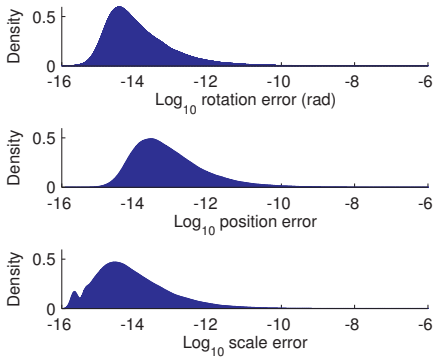


Figure 3: Distribution of numerical error in the computed pose and scale based on 10^5 random trials with four point-ray correspondences. The probability distribution function for each error dimension was estimated using kernel density estimation with a normal kernel function.

5. Evaluation

5.1. Numerical stability

To test the numerical stability of our solution, we tested the solution on synthetic data over 10^5 trials. Random camera configurations were generated with the ray origins $\mathbf{p}_i$ uniformly distributed in the volume $[-1,1] \times [-1,1] \times [-1,1]$, the anchor points $\mathbf{q}_i$ in the volume $[-1,1] \times [-1,1] \times [2,4]$, and the ray directions as $\mathbf{d}_i = (\mathbf{q}_i - \mathbf{p}_i)/||(\mathbf{q}_i - \mathbf{p}_i)||$. No transformation was applied to the input vectors, so that the correct solution for each trial was $\mathbf{R} = \mathbf{I}_3$, $\mathbf{t} = \mathbf{0}$ and $s = 1$. For each trial, the minimal four correspondences were used to produce solutions using our solver, and the angular error of a fifth correspondence was used to choose the best solution, denoted $\hat{\mathbf{R}}, \hat{\mathbf{t}}, \hat{s}$. We calculated three error measures: rotational error, the angular error between $\hat{\mathbf{R}}$ and $\mathbf{R}$; translational error, $||(-\mathbf{R}^\top \mathbf{t}) - (-\hat{\mathbf{R}}^\top \hat{\mathbf{t}})||$; and scale error, $|\hat{s} - s|$. The results of the experiment are shown in Figure 3. All error terms were below 1×10^{-11} in 96% of the trials.

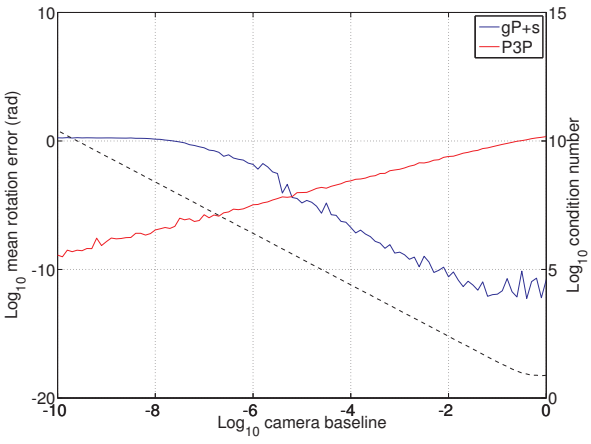


Figure 4: Relative accuracy of our generalized pose and scale solution and a pinhole camera pose solution which disregards the baseline between the camera rays. Each solution was tested over a range of baseline distances between 1×10^{-10} and 1. The condition number $\kappa(\mathbf{A}_{\mathbf{t}s})$ is plotted in a black dashed line.

5.2. Relation to the pinhole camera model

As discussed in Section 4.2, our solution becomes numerically unstable when the baseline between the rays is small relative to the distance to the anchor points. We investigated experimentally the change in accuracy of our solution as the baseline is reduced, in comparison with the P3P solution of Kneip *et al*. [16], which assumes that all rays meet at a common optical center. We generated synthetic four-point problems with anchor points uniformly distributed in the volume $[0,1] \times [0,1] \times [1,2]$. The camera rays were fixed as $\mathbf{p}_1 = \mathbf{p}_2 = \mathbf{0}^\top$, $\mathbf{p}_3 = \mathbf{p}_4 = [b\ 0\ 0]^\top$, where b is the chosen baseline distance.

For each trial, we computed the result of our gP+s algorithm on all four rays, and the result of standard P3P using the first three rays. We tested baseline values in the range $[1 \times 10^{-10}, 1]$ with 10^4 trials for each setting. Figure 4 shows a log-log plot of the rotational accuracy for both methods, as well as the condition number $\kappa(\mathbf{A}_{\mathbf{t},s})$; the translational accuracy (not shown) indicates a very similar pattern. The crossover-point in accuracy occurs when $\kappa \approx 10^{5.5}$.

5.3. Comparison on noisy synthetic data

We also compared the accuracy of our solution, on synthetic data, with two other possible solutions which are applicable in the overdetermined case, depending on the camera configuration. For the absolute orientation solution, each pair of rays from each point is used to triangulate a point, and then with the three pairs of corresponding points we compute the absolute orientation estimate using the method of Uchiyama [34]. For the P3P+s solution, we use the three rays in the first camera to calculate a pose estimate using the method of Kneip *et al*. [16]. The estimated pose of the first camera is taken as the rotation and translation, and the three observations in the second camera are used to produce a least-squares estimate of the scale. For our novel gP+s solution, all six rays are used together in our solver.

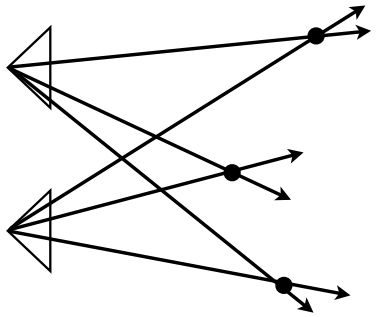
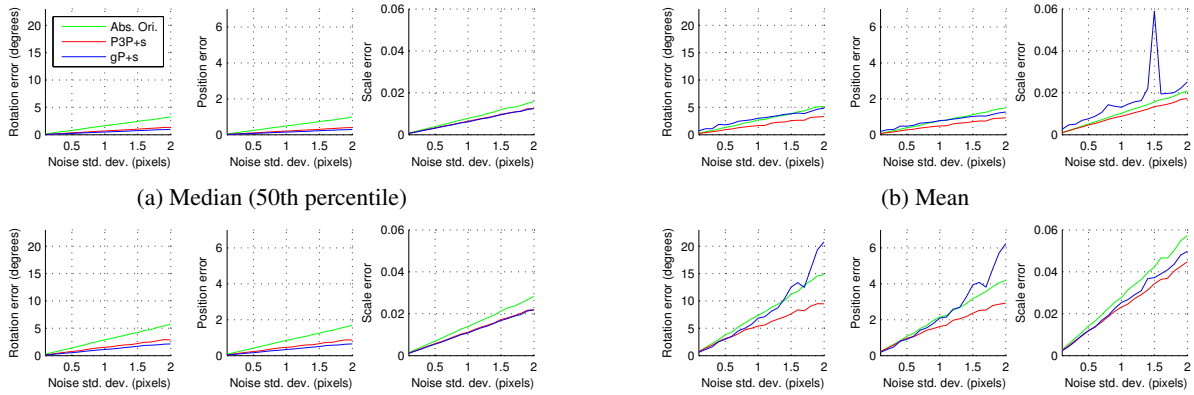


Figure 5: With this two camera, three point configuration, we tested three possible solutions to determine the pose and scale: triangulation of the points followed absolute orientation to align them; pose estimation of one camera followed by scale estimation with the other; and our direct solution using the six rays in an overdetermined system.

The conceptual advantage of our solution is that it uses all input rays to calculate all parameters together, whereas P3P+s separates the calculation of pose and scale, and absolute orientation transfers the 2D input error to an anisotropic 3D error by point triangulation.

To test all three methods using the same set of input rays, we used the three-point, two-camera setup illustrated in Figure 5. The points were uniformly distributed in the volume $[-10, 10] \times [-10, 10] \times [10, 20]$ and the cameras were placed at $[-1\ 0\ 0]^\top$ and $[1\ 0\ 0]^\top$ facing the +Z direction. We tested the accuracy of all solutions with Gaussian noise added to the projected point locations. We tested a range of noise levels from 0.1 to 2 pixels standard deviation, with 10^4 trials at each noise level.

The mean, median (50th percentile), 75th percentile, and 95th percentile error for each solution is plotted in Figure 6. There are certain samples for which each method produces a higher error than usual. These outliers affect the mean, but not the median. The mean is an unstable measure because of this. The median is of more interest, because it better reflects the quality of the solution when used in a random sampling framework like RANSAC, which is what is important in practice. Our novel solution has the best accuracy when comparing the 50th and 75th percentile error in rotation, translation and scale over the entire range of noise levels. The error of our solution is slightly worse at the 95th percentile; this means that our solution is better except in some 5% of cases.



(a) Median (50th percentile)

(b) Mean

(c) 75th percentile

(d) 95th percentile

Figure 6: Error in computed pose and scale with a three point, two camera configuration (six point-ray correspondences). Three methods are compared: point triangulation and absolute orientation (Abs. Ori.), P3P pose computation with one camera and scale from the other (P3P), and our generalized pose+scale solution (gP+s). A range of levels of Gaussian noise was added to the measurements, from 0.1 to 2 pixels standard deviation. Each noise level and method was tested with 10^4 random trials.

5.4. SLAM registration with real images

We tested the use of our minimal solver for registration of a monocular SLAM reconstruction with an existing SfM reconstruction. In previous work [35], we developed a real-time system for mobile camera tracking which registers a client-side SLAM map with a pre-made reconstruction on a server. The advantage of this system concept is that it reduces computation and storage costs on the mobile device, while providing globally registered tracking on the client device by way of loop closure of the local map with the global map on the server.

To test the accuracy of our solution in comparison to other possible map registration methods, we recorded several image sequences in a controlled indoor environment. An ART-2 tracking system provided precise optical tracking of a marker attached to the camera. One long image sequence was used in an offline, incremental SfM pipeline to produce a point cloud reconstruction of the environment, and to calibrate the transformation between the camera and the tracker coordinate systems. We then recorded twelve image sequences in the tracked environment, and processed them in a real-time capable keyframe-based SLAM system. This provided locally-referenced tracking of each image sequence. A top-down view of the SfM reconstruction and the paths calculated from the SLAM tracker and the actual ground-truth measurements from the ART-2 tracking system are shown in Figure 7. Note that the reconstructed area was a partition of a larger room which was separated by temporary walls. The walls were not set up perfectly perpendicular to each other, therefore the reconstruction looks skewed, although it is correct. Photographs of the room and sample images from the recorded sequences are shown in Figure 8.

We compared three procedures for registering a SLAM map to the global point cloud. As a preliminary step for each method, SIFT keypoints from each keyframe image are matched to the global point cloud using approximate nearest-neighbor matching and the ratio test [23]. Then the methods proceed as follows:

Absolute orientation Each global point aggregates a list of feature matches, and for each point with at least two matches, a consistent triangulation is found using the relative camera poses given by the SLAM system. Then the absolute orientation method is employed in a RANSAC loop to find a consistent registration of the triangulated points with the global points.

P3P+s Each keyframe image is localized separately using the P3P algorithm of Kneip *et al*. [16] in a PROSAC loop. Then for each camera, the median scale estimate over all feature matches in other cameras is taken. The pose+scale estimate with the most inliers is taken as the solution.

gP+s Our solution is used in a PROSAC loop with all feature matches from all keyframes.

For each image sequence and registration method, we computed the average positional error of all keyframes according to the ground truth given by the optical tracker. The results are summarized in Table 2. Our method gives the most accurate estimate in eight of the twelve sequences.

We also tested our system using sequences from the SLAM benchmark dataset of Sturm *et al*. [31]. In our tests, we used only the RGB camera sequences, without the depth images or other sensor data provided. We selected

Sequence	# Keyframes	Abs. Ori.	P3P+s	gP+s
office1	9	6.37	6.14	**6.12**
office2	9	8.09	7.81	**7.49**
office3	33	8.29	9.31	**6.78**
office4	9	4.76	4.48	**4.00**
office5	15	3.63	**3.42**	4.75
office6	24	**5.15**	5.23	5.91
office7	9	**6.33**	7.08	7.07
office8	11	4.72	4.85	**4.59**
office9	7	8.41	8.44	**6.65**
office10	23	**5.88**	6.60	**5.88**
office11	58	5.19	**4.85**	6.74
office12	67	5.53	5.20	**4.86**
fr1/desk	121	13.91	13.22	**12.16**
fr1/desk2	50	6.95	**5.37**	5.83

Table 2: Average keyframe registration accuracy, in centimeters. The *office* sequences were created in small office setting with an ART-2 optical tracker providing ground truth pose measurements. They were registered against a structure-from-motion point cloud reconstruction of the scene. The *desk* sequences sequences are from the benchmark dataset of Sturm *et al*. [31]. These sequences were registered against a point cloud reconstruction made using the *fr1/room* sequence.

the *fr1/room* sequence to produce a reference point cloud using our batch SfM pipeline. The ground truth measurements from a Vicon tracker were included in the bundle adjustment error term, in order to align the reconstruction with the ground truth coordinate system. The sequences *fr1/desk* and *fr1/desk2* are two other image sequences which view the same scene with handheld camera motion. Because these sequences had very fast movement and significant motion blur, we could not successfully process them in our monocular SLAM system. Instead, we again used our batch SfM pipeline to process the two *desk* sequences, without the ground truth measurements added. Finally, we tested all three SLAM registration methods on these two sequences for registering with the reference *room* reconstruction. The results are summarized in the last two rows of Table 2.

6. Conclusion

In this work, we proposed a new absolute pose-and-scale problem for a generalized camera model. Our solver produces an accurate estimate in both the minimal and overdetermined cases. Furthermore, it is efficient enough to be applied in a random sampling framework for robust estimation with noisy measurements.

Through analysis on synthetic datasets, we showed the numerical stability of the solution, the relation to the pinhole camera model and detection of this degenerate case,

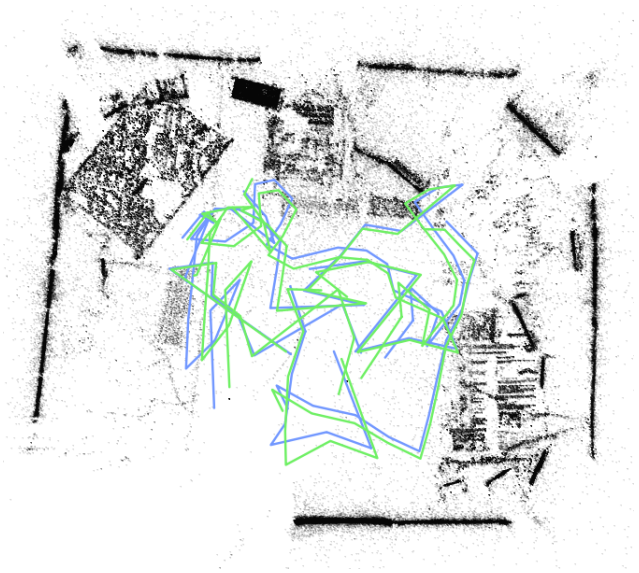


Figure 7: SLAM Keyframes from the image sequences (green) after registration with the 3D point cloud (black points). The ground truth paths (blue) were measured with an ART-2 optical tracker.

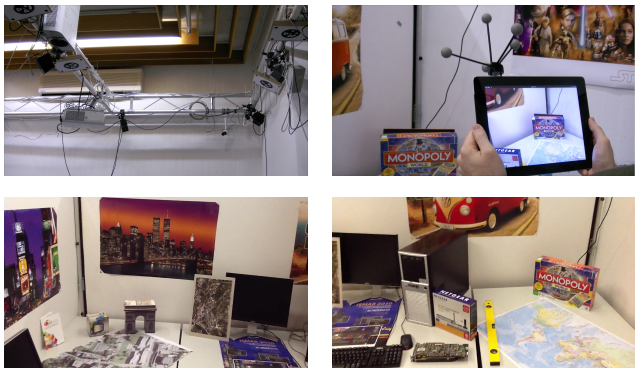


Figure 8: Example photographs from our ART2 setup and the indoor tracking area.

and a comparison with other possible special-case solutions. Finally, our tests on real-world image sequences show the usefulness of our method for loop closure and registration of SLAM and structure-from-motion results.

References

[1] M. Brown, R. Hartley, and D. Nistér. Minimal solutions for panoramic stitching. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007. 2

[2] M. Bujnak, Z. Kúkelová, and T. Pajdla. A general solution to the P4P problem for camera with unknown focal length. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 2

[3] M. Bujnak, Z. Kukelova, and T. Pajdla. Making minimal solvers fast. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1506–1513, 2012. 2, 4

[4] C.-S. Chen and W.-Y. Chang. On pose recovery for generalized visual sensors. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 26(7):848–861, 2004. 1, 2

[5] O. Chum and J. Matas. Matching with PROSAC - progressive sample consensus. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 220–226 vol. 1, 2005. 2

[6] L. A. Clemente, A. J. Davison, I. Reid, and J. Neira. Mapping large loops with a single hand-held camera. In *Robotics: Science and Systems*, 2007. 2

[7] J. Courchay, A. Dalalyan, R. Keriven, and P. Sturm. Exploiting loops in the graph of trifocal tensors for calibrating a network of cameras. In *European Conference on Computer Vision (ECCV)*, volume 6312, pages 85–99, 2010. 2

[8] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms*. Springer, 3rd edition, 2007. 3

[9] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 29(6):1052–1067, 2007. 2

[10] E. Eade and T. Drummond. Unified loop closing and recovery for real time monocular slam. In *British Machine Vision Conference (BMVC)*, 2008. 2

[11] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. of the ACM*, 24(6):381–395, June 1981. 2

[12] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2nd edition, 2004. 4

[13] B. Horn, H. M. Hilden, and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 5(7):1127–1135, 1988. 2

[14] M. Klopschitz, C. Zach, A. Irschara, and D. Schmalstieg. Generalized detection and merging of loop closures for video sequences. In *Int. Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, 2008. 2

[15] L. Kneip, P. T. Furgale, and R. Siegwart. Using multi-camera systems in robotics: Efficient solutions to the NPnP problem. In *IEEE Int. Conference on Robotics and Automation (ICRA)*, May 2013. 2

[16] L. Kneip, D. Scaramuzza, and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, USA, June 2011. 1, 4, 5, 6

[17] Z. Kúkelová, M. Bujnak, and T. Pajdla. Automatic Generator of Minimal Problem Solvers. In *European Conference on Computer Vision (ECCV)*, pages 302–315. Springer, 2008. 2, 3

[18] Z. Kukélová, M. Bujnak, and T. Pajdla. Closed-form solutions to minimal absolute pose problems with known vertical direction. In *Asian Conference on Computer Vision (ACCV)*, pages 216–229. Springer, 2011. 2

[19] Z. Kúkelová and T. Pajdla. A minimal solution to the autocalibration of radial distortion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–7, 2007. 2

[20] V. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: An accurate $O(n)$ solution to the PnP problem. *Int. Journal Computer Vision (IJCV)*, 2009. 2, 4

[21] H. Li. A simple solution to the six-point two-view focal-length problem. In *European Conference on Computer Vision (ECCV)*, volume 4, pages 200–213. Springer, 2006. 2

[22] H. Li and R. Hartley. Five-point motion estimation made easy. In *Int. Conference on Pattern Recognition (ICPR)*, volume 1, pages 630–633, 2006. 2

[23] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal Computer Vision (IJCV)*, 60(2):91–110, Nov. 2004. 6

[24] D. Nistér. An efficient solution to the five-point relative pose problem. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2003. 2, 3

[25] D. Nistér. A minimal solution to the generalised 3-point pose problem. In *CVPR*, 2004. 1, 2

[26] D. Nistér and H. Stewénius. A minimal solution to the generalised 3-point pose problem. *Journal of Mathematical Imaging and Vision*, 27(1):67–79, 2007. 4

[27] W. H. Press, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computing, Third Edition*. Cambridge University Press, 2007. 3

[28] G. Schweighofer and A. Pinz. Globally optimal $O(n)$ solution to the PnP problem for general camera models. In *British Machine Vision Conference (BMVC)*, pages 1–10, 2008. 2

[29] H. Stewénius, D. Nistér, F. Kahl, and F. Schaffalitzky. A minimal solution for relative pose with unknown focal length. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 789–794, 2005. 2

[30] H. Strasdat, J. Montiel, and A. Davison. Scale drift-aware large scale monocular SLAM. In *Robotics: Science and Systems*, 2010. 2

[31] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *IEEE Int. Conference on Intelligent Robots and System (IROS)*, Oct. 2012. 6, 7

[32] S. Tariq and F. Dellaert. A multi-camera 6-DOF pose tracker. In *Int. Symposium on Mixed and Augmented Reality (ISMAR)*, pages 296–297, 2004. 2

[33] S. Thrun and M. Montemerlo. The GraphSLAM algorithm with applications to large-scale mapping of urban structures. *International Journal on Robotics Research*, 25(5/6):403–430, 2005. 2

[34] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 1991. 2, 5

[35] J. Ventura, C. Arth, G. Reitmayr, and D. Schmalstieg. Global localization from monocular SLAM on a mobile phone. In *IEEE Virtual Reality*, 2014. 6

[36] B. Williams, M. Cummins, J. Neira, P. Newman, I. Reid, and J. Tardós. An image-to-map loop closing method for monocular SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008. 2