Calibrated and Partially Calibrated Semi-Generalized Homographies Supplementary Material

Snehal Bhayani[1] Torsten Sattler[2] Daniel Barath[3] Patrik Beliansky[4]
Janne Heikkilä [1] Zuzana Kukelova[5]
[1]Center for Machine Vision and Signal Analysis, University of Oulu, Finland
[2]Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague
[3]Computer Vision and Geometry Group, Department of Computer Science, ETH Zürich
[4]Faculty of Mathematics and Physics, Charles University, Prague
[5]Visual Recognition Group, Faculty of Electrical Engineering, Czech Technical University in Prague

In this supplementary material, we provide additional details about various solvers for the calibrated as well as partially calibrated case (unknown focal length) that were not included in the main paper due to a lack of space. Moreover, we discuss the potential "$g_{33} = 0$" degeneracy for partially calibrated camera solvers. This is followed by details on elimination ideals used for the simplifications of polynomial systems in both considered cases, as well as the Macaulay2 code for generating these ideals. The last two sections provide more details about the synthetic and real experiments. The supplementary material follows the notations and conventions used in the main paper.

1. Solver Sizes w/o Variable Transformation

In all proposed solvers ($\texttt{sH5}_\mathbf{2}$, $\texttt{sH4.5}_\mathbf{2}$, $\texttt{sH5}_\mathbf{3}$, $\texttt{sH4.5}_\mathbf{3}$, $\texttt{sH5f}_\mathbf{3}$, and $\texttt{sHf5}_\mathbf{3}$) except the $\texttt{sH5}_\mathbf{4}$ solver, we simplify the systems of polynomial equations using a variable transformation followed by an elimination ideal trick. This transformation was described in Sec. 2.2 in the main paper. For comparison and for showing the usefulness of these steps, we also studied solvers without the variable transformation.

The elimination ideal trick used to eliminate unknowns from the ideal I defined by Eq. (9) in the main paper (see Sec. 2.2 in the main paper) results in 10 constraints (4 for the unknown focal length case) in 12 unknowns from $\mathtt{G}$ and $\mathbf{m}$. These are generators of the elimination ideal I_1 described in the main paper. After a null-space reparameterization of $\mathtt{G}$ and $\mathbf{m}$ using the linear equations (Eq. (8) in the main paper), we can transform these equations to 10 polynomial equations in 2 unknowns, respectively 3 unknowns for the 4.5 point correspondences in the calibrated case. Such systems can be solved, *e.g.*, using the automatic generator of Gröbner basis solvers [5, 6].

For the case of calibrated cameras using all linear constraints from 5 correspondences, we have to solve a system of two polynomial equations in two unknowns. Note that here, similar to the $\texttt{sH5}_\mathbf{2}$ and $\texttt{sH5}_\mathbf{3}$ solvers presented in the main paper, we have an over-constrained system and we are using only a subset (2 equations) from all 10 constraints defining the ideal I_1. Using the automatic generator [6], we can create a solver which performs a G-J elimination of a matrix of size 12×20 and has up to 8 real solutions for no more than two points coming from the same camera. For the configuration where three points are coming from the same camera, the solver generated using [6] performs a G-J elimination of a matrix of size 4×10 and returns up to 6 real solutions.

For 4.5 point correspondences, the null-space reparameterization of $\mathtt{G}$ and $\mathbf{m}$ results in 10 equations (generators of I_1) in three unknowns. Since we dropped one constraint from Eq. (8) from the main paper, we now have a well-defined system of polynomial equations that can be again solved using the automatic generator [6]. In this case, we obtain a solver which performs a G-J elimination of a matrix of size 33×49 and returns up to 16 real solutions, for both the case where no more than two points are coming from the same camera and the case where exactly three points are coming from the same camera.

For partially calibrated cameras, after the null-space reparameterization we obtain 4 polynomial equations in two variables. The solver generated using [6] has to perform G-J elimination of a matrix of size 6×12 and returns up to 6 real solutions for the case where at most 2 points come from the same camera. For three points coming from the same camera we generate a solver that performs a G-J elimination of a matrix of size 3×7 and returns up to 4 real solutions.

Note, that all solvers without the variable transformation are larger and therefore slower than their counterparts that use the variable transformation.

2. Solvers using 4.5 Correspondences

When the camera $\mathcal{P}$ is calibrated, 5 point correspondences result in an over-constrained formulation which can be solved in two ways. The first method is described in the main paper. This method uses 10 linear constraints from all 5 point correspondences (Eq. (8) from the main paper). However, it does not use all constraints resulting from the structure of the matrices, *i.e.*, all generators of the ideal I'_2.

Here, we describe the second method. This method uses 4.5 point correspondences, *i.e.*, 9 constraints out of the 10 available constraints of the form of Eq. (8) in our main paper. In this case, we perform the same coordinate system transform as the one that we use for the first approach, *i.e.*, in solvers $\mathtt{sH5_2}$ and $\mathtt{sH5_3}$. In this coordinate system transform, two constraints coming from the first point correspondence result in simple constraints on the matrix $\mathtt{G}$, *i.e.*, $g_{13} = 0,\ g_{23} = 0$.

Assuming $g_{33} \neq 0$, we divide the remaining 7 equations (from Eq. (8) in the main paper) by g_{33}, leading to 7 equations in 9 unknowns, $\varepsilon' = \{g'_{11}, g'_{12}, g'_{21}, g'_{22}, g'_{31}, g'_{32}, m'_1, m'_2, m'_3\}$. These equations can be written in matrix form

$$\mathtt{C}\mathbf{b} = 0 \ , \tag{1}$$

where $\mathtt{C}$ is a 7×10 coefficient matrix and $\mathbf{b}$ is a 10×1 vectorized form of the set of $\varepsilon' \cup \{1\}$.

$\mathtt{sH4.5_2}$ solver: In the first case, we assume that no more than two correspondences come from the same camera $\mathcal{G}_i$. Now, matrix $\mathtt{C}$ in (1) has a three dimensional null-space $\{\mathbf{b_1}, \mathbf{b_2}, \mathbf{b_3}\}$. A solution to $\mathbf{b}$ can be obtained as a linear combination $\mathbf{b} = \gamma_1\mathbf{b_1} + \gamma_2\mathbf{b_2} + \gamma_3\mathbf{b_3}$. Using $b_{10} = 1$, we can express γ_3 as a linear polynomial in γ_1 and γ_2. Hence the variables in ε' can be parameterized as linear polynomials of γ_1 and γ_2. This parameterization can be substituted into the generators of the ideal I'_2 for the calibrated case. This leads to 5 polynomials $e_i(\gamma_1, \gamma_2)$, each of degree 5. This system has up to 16 solutions. We use the automatic generator [6] to solve this system, resulting in a solver of size 11×27. Similar to the $\mathtt{sH5_2}$ solver, we next extract the solution to g_{33}, and then decompose the homography matrix $\mathtt{H}$ to obtain solutions for the relative poses $\mathtt{R}$ and $\mathbf{t}$. As mentioned in our main paper, we ensure that $\mathtt{H}$ is decomposed such that $det(\mathtt{R}) = 1$.

$\mathtt{sH4.5_3}$ solver: In this case, we assume that there are three correspondences coming from the same camera $\mathcal{G}_i$. Let us assume, w.l.o.g., that the points $\mathbf{q}_{i2}$ and $\mathbf{q}_{i3}$ are observed in camera $\mathcal{G}_1$, *i.e.*, $i = 1$. This is the same camera that observes the point $\mathbf{q}_{11}$. The remaining two points can be observed by one camera $\mathcal{G}_j \neq \mathcal{G}_1$ or by two different cameras $\mathcal{G}_j \neq \mathcal{G}_k \neq \mathcal{G}_1$. Our approach in this case is analogous to that of the $\mathtt{sH4.5_2}$ solver. We select 9 constraints from Eq. (8) such that the first three points $\mathbf{q}_{i1}$, $\mathbf{q}_{i2}$ and $\mathbf{q}_{i3}$ contribute only 5 constraints. This ensures that even if these three points are collinear, the matrix $\mathtt{C}$ in (1) has rank 7.

The polynomial system $e_i(\gamma_1, \gamma_2),\ i = 1, \dots 5$ has up to 12 solutions. Using the automatic generator [6], we can generate a solver of size 23×35. The next steps extract the solution to g_{33}, and then decompose the homography matrix $\mathtt{H}$ to obtain solutions for $\mathtt{R}$ and $\mathbf{t}$.

3. Unknown Focal Length Solvers

In this case the focal length of the pinhole camera $\mathcal{P}$ is unknown and we assume its calibration matrix to be of the form $\mathtt{K} = \text{diag}(f, f, 1)$. Thus. we have 10 DOF, 3 for each of $\mathtt{R}$, $\mathbf{t}$ and $\widetilde{\mathbf{n}}$, and 1 for f. Therefore, we need 5 2D-2D correspondences $\mathbf{p}_j \leftrightarrow \mathbf{q}_{ij},\ j = 1, \dots 5$, to solve this problem. Without loss of generality, we can assume that the first point correspondence is observed in camera $\mathcal{G}_1$, *i.e.*, $i = 1$. For $i = 1$, we have $\mathbf{t}_{\mathbb{G}_1} = [0, 0, 0]^\top$. Moreover, w.l.o.g., we can pre-rotate the local coordinate systems of $\mathcal{P}$ and $\mathcal{G}_1$ such that $\mathbf{p}_1 = [1, 0, 1]^\top$ and $\mathbf{q}_{11} = [0, 0, 1]^\top$. This simplifies the equations, and after substituting into Eq. (8) in the main paper, we have

$$g_{13} = -g_{11},\ g_{23} = -g_{21} \ . \tag{2}$$

The remaining four 2D-2D correspondences give us 8 linearly independent constraints of the form of Eq. (8) in the main paper. We assume that $g_{33} \neq 0$ and divide these equations by g_{33}. This, together with (2), transforms these equations into non-homogeneous equations in 9 unknowns, *i.e.*, equations containing unknowns $\varepsilon' = \{g'_{11}, g'_{12}, g'_{21}, g'_{22}, g'_{31}, g'_{32}, m'_1, m'_2, m'_3\}$. We can rewrite these equations in matrix form

$$\mathtt{C}\mathbf{b} = 0 \ , \tag{3}$$

where $\mathtt{C}$ is a 8×10 coefficient matrix and $\mathbf{b}$ is a 10×1 vectorized form of the set of $\varepsilon' \cup \{1\}$. Next, we propose solvers for two different situations, one where a maximum of two correspondences come from the same camera and another where three correspondences come from the same camera. For the case where four correspondences come from the same camera $\mathcal{G}_i$, this section contains a proof that such a formulation cannot be solved using just one more correspondence coming from a camera $\mathcal{G}_j,\ j \neq i$.

$\mathtt{sH5f_2}$ solver: Assuming that no more than two correspondences are coming from the same camera $\mathcal{G}_i$, the matrix $\mathtt{C}$ in (3) has a two dimensional null-space $\{\mathbf{b_1}, \mathbf{b_2}\}$. A solution to $\mathbf{b}$ can be obtained as a linear combination $\mathbf{b} = \gamma_1\mathbf{b_1} + \gamma_2\mathbf{b_2}$. Using the constraint $b_{10} = 1$, we can express γ_2 as a linear polynomial in γ_1. Hence the variables in ε' can be parameterized as linear polynomials of γ_1. Substituting such a parameterization into the generator e of the ideal I'_2 (described in Sec. 2.2 in our main paper) yields a

univariate polynomial $e(\gamma_1)$ of degree five. This polynomial can be efficiently solved using Sturm sequences [3].

The next step is to extract solutions to g_{33} and f. Writing $\mathtt{G} = \mathtt{R}\mathtt{K}^{-1} - \mathbf{t}\mathbf{m}^\top$, we obtain a set of polynomial constraints. By variable elimination and substitutions, we obtain the solution to g_{33}, unique up to a sign, which is fixed by constraining the solution of the plane vector $\widetilde{\mathbf{n}}$ so that the corresponding 3D point in $\mathcal{P}$ is in the front of the camera. Moreover, using a similar approach we obtain a unique solution to the focal length f. Solutions to $\mathtt{G}$ as well as $\mathbf{m}$ can be extracted from the solutions to ε' and g_{33}. Knowing f, we can compute $\mathtt{H} = \mathtt{G}\mathtt{K}$ as well as $\widetilde{\mathbf{n}} = \mathtt{K}\mathbf{m}$. Decomposing the homography matrix $\mathtt{H}$ leads to set of relative poses $\mathtt{R}$ and $\mathbf{t}$. As mentioned in our main paper, we ensure that $\mathtt{H}$ is decomposed such that $det(\mathtt{R}) = 1$.

sH5f$_3$ solver: If there are three 2D-2D point correspondences coming from the same camera $\mathcal{G}_i$, the situation is a bit different. Let us assume, w.l.o.g., that the points $\mathbf{q}_{i2}$ and $\mathbf{q}_{i3}$ are observed in camera $\mathcal{G}_1$, *i.e.*, $i = 1$. This is the same camera that observes the point $\mathbf{q}_{11}$. The remaining two points can be observed by one camera $\mathcal{G}_j \neq \mathcal{G}_1$ or by two different cameras $\mathcal{G}_j \neq \mathcal{G}_k \neq \mathcal{G}_1$. In this case, G-J elimination of the matrix $\mathtt{C}$ in (3) leads to a matrix of a special form

$$\begin{bmatrix} \mathtt{I}_{6\times6} & \mathtt{0}_{6\times2} & \mathbf{0}_{6\times1} & \mathbf{c}_{6\times1} \\ \mathtt{0}_{2\times6} & \mathtt{I}_{2\times2} & \mathbf{d}_{2\times1} & \mathbf{e}_{2\times1} \end{bmatrix} \mathbf{b} = \mathbf{0} \ , \tag{4}$$

where the indices of the matrices and vectors indicate their sizes. Since $\mathbf{b} = [g'_{11}, g'_{12}, g'_{21}, g'_{22}, g'_{31}, g'_{32}, m'_1, m'_2, m'_3, 1]^\top$, the first six rows of (4) directly give us a solution to g'_{kl}. The last two rows can be used to express m'_1, m'_2 as a linear function of m'_3. Substituting these expressions for m'_1, m'_2 and the extracted solution to g'_{kl} into the polynomial e, which is the generator of the ideal I'_2 for the unknown focal length case (described in Sec. 2.2 in our main paper) gives us a univariate polynomial in m'_3 of degree three. The remaining steps are similar to that of the **sHf5$_2$** solver.

sH5f$_4$ case: In this case, we have four 2D-2D point correspondences observed by the same camera $\mathcal{G}_i$. In such a situation, it is impossible to solve for both the focal length and the scale of the translation using only one additional correspondence from a camera $\mathcal{G}_k$, $k \neq i$[1]. The following proof is algebraic, *i.e.*, we show that all available constraints do not define a zero-dimensional ideal. In other words, we show that the system of equations is under-constrained and has an infinite number of solutions.

Lemma 3.1. *When the focal length of the camera $\mathcal{P}$ is unknown and we have four 2D-2D point correspondences, $\mathbf{p}_j \leftrightarrow \mathbf{q}_{ij}, j = 1, \ldots, 4$, observed by the same camera $\mathcal{G}_i$, an additional point correspondence $\mathbf{p}_j \leftrightarrow \mathbf{q}_{kj}, j = 5$, observed by a different camera $\mathcal{G}_k$ does not provide a sufficient number of constraints to estimate both the focal length and the scale of the translation.*

Proof. Each point correspondence, $\mathbf{p}_j \leftrightarrow \mathbf{q}_{ij}$, leads to an equation of the form (Eq. (8) in the main paper),

$$[\mathbf{q}_{ij}]_\times (\mathtt{G}\mathbf{p}_j + (\mathbf{m}^\top \mathbf{p_j})\mathbf{t}_{\mathcal{G}_i}) = 0 \ . \tag{5}$$

Due to the rank-2 skew symmetric matrix $[\mathbf{q}_{ij}]_\times$, only two out of the three equations in the matrix equation in (5) are linearly independent. This means that each point correspondence yields 2 linear homogeneous constraints in the elements of $\mathtt{G}$ and $\mathbf{m}$.

We assume that $g_{33} \neq 0$ (see Sec. 2.2 in our main paper) which allows us to divide these constraints by g_{33}. The resulting variables are $\varepsilon' = \{g'_{11}, g'_{12}, g'_{13}, g'_{21}, g'_{22}, g'_{23}, g'_{31}, g'_{32}, m'_1, m'_2, m'_3\}$ which gives us 11 DOF.

In the main paper (see Sec. 2.2 in the main paper for details), it was shown that after this variable transformation there is one additional constraint in variables ε'. This constraint e is the generator of the elimination ideal I'_2 that was created using all available constraints on $\mathtt{G}$ and $\mathbf{m}$.

Let us assume without loss of generality that $i = 1$ and $\mathbf{t}_{\mathcal{G}_1} = [0, 0, 0]^\top$. Hence the first 4 point correspondences give us 8 linearly independent linear constraints in $\{g'_{11}, g'_{12}, g'_{13}, g'_{21}, g'_{22}, g'_{23}, g'_{31}, g'_{32}\}$. This leaves us with 3 DOF, *i.e.*, $\{m'_1, m'_2, m'_3\}$.

The two constraints of the form (5) obtained from the fifth point correspondence coming from camera $\mathcal{G}_k$ ($k \neq i = 1$), as well as the polynomial e are the only remaining constraints on $\{m'_1, m'_2, m'_3\}$. However, in this case, the two constraints from the fifth point correspondence are linearly dependent, which we show next.

Let us denote $\mathtt{G}' = \mathtt{G}/g_{33}$ and $\mathbf{m}' = \mathbf{m}/g_{33}$. With this notation Eq. (5) for the fifth correspondence (*i.e.*, $j = 5$) becomes

$$[\mathbf{q}_{k5}]_\times (\mathtt{G}'\mathbf{p}_5 + (\mathbf{m}'^\top \mathbf{p_5})\mathbf{t}_{\mathcal{G}_k}) = 0 \ . \tag{6}$$

However, having extracted the solutions to $\{g'_{11}, g'_{12}, g'_{13}, g'_{21}, g'_{22}, g'_{23}, g'_{31}, g'_{32}\}$, $\mathtt{G}'$ is fixed. Thus we can rewrite this equation as $\mathbf{a}_1 + \lambda[\mathbf{q}_{k5}]_\times(\mathbf{t}_{\mathcal{G}_k}) = 0$ where $\mathbf{a}_1 = [\mathbf{q}_{k5}]_\times \mathtt{G}'\mathbf{p}_5$ is a known 3×1 vector and $\lambda = (\mathbf{m}'^\top \mathbf{p_5})$ in a linear function in $\{m'_1, m'_2, m'_3\}$. This implies that the fifth correspondence imposes only one constraint on λ and hence on $\{m'_1, m'_2, m'_3\}$.

Altogether with the constraint e we have only two linearly independent polynomial equations in three variables $\{m'_1, m'_2, m'_3\}$ and hence an under-constrained system of polynomials that is not solvable.

□

[1]Note that this also holds for $k = i$. Moreover, in this case, it is not possible to recover the scale even after fixing the focal length f.

4. Degeneracy for Focal Length Solvers

To obtain efficient solvers, we performed a variable substitution that was assuming that entry $g_{33} \neq 0$ (for more details, see Sec. 2.2 in the main paper). This substitution introduces a potential degeneracy into our solvers, *i.e.*, a degeneracy for homographies with entry $g_{33} = 0$. Our chosen coordinate system transformation allowed us to circumvent this situation for the calibrated solvers. However, for the partially calibrated case, this degeneracy can potentially lead to numerical instabilities of the solver when dealing with homographies with entry g_{33} close to zero. Yet, as we show next, our solvers for partially calibrated cameras are numerically stable even in situations that are very close to this degenerate case. Thus, this degeneracy is not an issue in practice.

To test the numerical stability of our solvers in the close-to-degenerate case, we created various synthetic scenes by varying the magnitude of g_{33} w.r.t. normalized `G` and gradually bringing it closer to 0. We evaluated the performance of the proposed solvers **sH5f$_2$** and **sH5f$_3$** on these scenes. In Fig. 1, we plot the error in the estimated rotation w.r.t. the ground truth as the magnitude of g_{33} gradually approaches 0. For this experiment, we implemented **sH5f$_2$** using Sturm sequences [3]. For **sH5f$_3$**, we implemented two variants, one using Sturm sequences and the other one computing eigenvalues of the companion matrix.

We observe that **sH5f$_2$** maintains its stability even as g_{33} reaches a magnitude close to 10^{-11}. In case of the **sH5f$_3$** solver, our implementation based on Sturm sequences begins to experience instability w.r.t. decreasing g_{33}. In contrast, our companion matrix-based implementation remains stable. Since **sH5f$_3$** computes the roots of a polynomial of degree three, there is no a significant difference between the efficiency of the Sturm-based solver and the companion matrix-based one. Therefore, we can directly use the companion matrix-based solver that is numerically stable even in close-to-degenerate situations.

Moreover, the proposed solvers are usually used inside a RANSAC framework. Thus, after observing numerical instabilities caused by $g_{33} \approx 0$, one can always switch to a solver that is based on a different variable substitution, *i.e.*, a solver which assumes that a different entry $g_{kl} \neq 0$. Such solvers have the same structure and solve a similar polynomial of degree 5 respectively 3, as the proposed solvers.

Our assumption that $g_{33} \neq 0$ helped to create more efficient solvers. Yet, we have also considered a formulation where such an assumption is not needed. This formulation directly eliminates the mentioned degenerate configuration and it results in solving a system of four polynomial equations in two unknowns (see Sec. 1). We use the automatic generator [6] to construct a Gröbner basis solver for this system of equations which performs a Gauss-Jordan elimination of a 3×7 matrix and an eigenvalue decomposition of a 4×4 matrix. Such a solver is slightly slower than the presented **sH5f$_3$**.

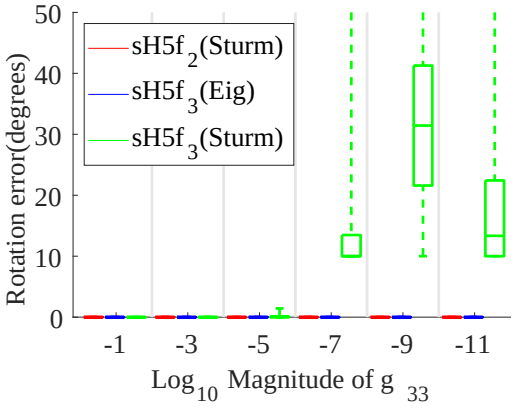

Figure 1. Rotation error in degrees of the proposed unknown focal length solvers in the close-to-degenerate case, *i.e.*, when $g_{33} \to 0$. We implemented solvers **sH5f$_2$** and **sH5f$_3$** using Sturm sequences and, also, **sH5f$_3$** based on the eigenvalues of the companion matrix.

5. Macaulay2 Code for Elimination Ideals

The elimination ideals I_1 and I_2' considered in Sec. 2.2 of our main paper have been computed using the computer algebra system `Macaulay2` [2].

5.1. Unknown Focal Length Solvers

For the unknown focal length solvers, we use the following lines of code to generate I_1:

```
R = QQ[g11,g12,g13,g21,g22,g23,g31,g32,g33,
r11,r12,r13,r21,r22,r23,r31,r32,r33,
m1,m2,m3, t1,t2,t3, w ];
-- The rotation matrix
Rot = matrix {{r11,r12,r13},
              {r21,r22,r23},
              {r31,r32,r33}};
-- The plane vector
M = matrix {{m1,m2,m3}};
-- The translation vector
t = matrix{{t1},{t2},{t3}};
K = matrix {{1_R,0_R,0_R},{0_R,1_R,0_R},
            {0_R,0_R,w}};
E = matrix {{1_R,0_R,0_R},{0_R,1_R,0_R},
            {0_R,0_R,1_R}};
-- Semi-generalized homography constraint
G =  Rot * K - t * M ;

I  = ideal(submatrix(G,{0},{0})-g11,
        submatrix(G,{0},{1})-g12,
        submatrix(G,{0},{2})+g11,
        submatrix(G,{1},{0})-g21,
        submatrix(G,{1},{1})-g22,
```

```
        submatrix(G,{1},{2})+g21,
        submatrix(G,{2},{0})-g31,
        submatrix(G,{2},{1})-g32,
        submatrix(G,{2},{2})-g33) +
        minors(1,transpose(Rot)*Rot-E)+
        minors(1,Rot*transpose(Rot)-E);

-- Eliminate translation and focal length
I1 = eliminate({t1,t2,t3,w}, I);

-- Eliminate rotation matrix
I1 = eliminate({ r11,r12,r13,
                 r21,r22,r23,
                 r31,r32,r33}, I1);

betti mingens I1
```

Referring to Sec. 2.2 in our main paper, we note that each variable `gkl`, $1 \leq k,l \leq 3$ corresponds to the variable g_{kl} and `mk`, $1 \leq k \leq 3$ denotes the variables m_k. The output of this code snippet shows that I_1 is generated by four polynomials, of which three are of degree 4 and one is of degree 5.

The following code then performs the variable substitution, where each variable `gkl`, $1 \leq k,l \leq 3$ after the variable substitution denotes g'_{kl} and `mk`, $1 \leq k \leq 3$ denotes the variables m'_k:

```
I1prime = sub(I1,
            {g11=> g33*g11, g12=> g33*g12,
            g21=> g33*g21, g22=> g33*g22,
            g31=> g33*g31, g32=> g33*g32,
            m1=> g33*m1, m2=> g33*m2,
            m3=> g33*m3 });

f0 = factor I1prime_0; f0 = f0#1#0;
f1 = factor I1prime_1; f1 = f1#1#0;
f2 = factor I1prime_2; f2 = f2#1#0;
f3 = factor I1prime_3; f3 = f3#1#0;
I1prime = ideal({f0,f1,f2,f3});
I2prime  = eliminate({g33}, I1prime);
e = mingens I2prime
betti mingens I2prime
```

The output of this code are the generators of the elimination ideal I'_2, which is in this case only one polynomial e, of degree 5 of the following form:

$$\begin{gathered} e = g_{11}g_{12}m_1^3 + g_{21}g_{22}m_1^3 - g_{11}^2 m_1^2 m_2 - \\ g_{21}^2 m_1^2 m_2 + g_{11}g_{12}m_1 m_2^2 + g_{21}g_{22}m_1 m_2^2 - \\ g_{11}^2 m_2^3 - g_{21}^2 m_2^3 + g_{11}g_{12}m_1^2 m_3 + g_{21}g_{22}m_1^2 m_3 + \\ g_{31}g_{32}m_1^2 m_3 - g_{11}^2 m_1 m_2 m_3 + g_{12}^2 m_1 m_2 m_3 - g_{21}^2 m_1 m_2 m_3 + \\ g_{22}^2 m_1 m_2 m_3 - g_{31}^2 m_1 m_2 m_3 + g_{32}^2 m_1 m_2 m_3 - \\ g_{11}g_{12}m_2^2 m_3 - g_{21}g_{22}m_2^2 m_3 - g_{31}g_{32}m_2^2 m_3 - \\ g_{32}m_1^3 + g_{31}m_1^2 m_2 - g_{32}m_1 m_2^2 + g_{31}m_2^3 \ . \end{gathered} \tag{7}$$

5.2. Calibrated Solvers

In the calibrated case, `K` is the identity matrix and $w = 1$. We use the following lines of code to generate I_1:

```
R = QQ[g11,g12,g13,g21,g22,g23,g31,g32,g33,
r11,r12,r13,r21,r22,r23,r31,r32,r33,
m1,m2,m3, t1,t2,t3 ];
-- The rotation matrix
Rot = matrix {{r11,r12,r13},
              {r21,r22,r23},
              {r31,r32,r33}};
-- The plane vector
M = matrix {{m1,m2,m3}};
-- The translation vector
t = matrix{{t1},{t2},{t3}};
E = matrix {{1_R,0_R,0_R},{0_R,1_R,0_R},
            {0_R,0_R,1_R}};
K = E;
-- Semi-generalized homography constraint
G =  Rot * K - t * M ;
I  = ideal(submatrix(G,{0},{0})-g11,
        submatrix(G,{0},{1})-g12,
        submatrix(G,{0},{2}),
        submatrix(G,{1},{0})-g21,
        submatrix(G,{1},{1})-g22,
        submatrix(G,{1},{2}),
        submatrix(G,{2},{0})-g31,
        submatrix(G,{2},{1})-g32,
        submatrix(G,{2},{2})-g33) +
        minors(1,transpose(Rot)*Rot-E)+
        minors(1,Rot*transpose(Rot)-E);

-- Eliminate translation and focal length
I1 = eliminate({t1,t2,t3}, I);
-- Eliminate rotation matrix
I1 = eliminate({ r11,r12,r13,
                 r21,r22,r23,
                 r31,r32,r33}, I1);

betti mingens I1
```

The output of this code snippet shows that in the calibrated case, I_1 is generated by ten polynomials, six of which are of degree 4, three are of degree 5, and one is of degree 6. The following code then performs the variable substitution presented in Sec. 2.2 of the main paper:

```
I1prime = sub(I1,
            {g11=> g33*g11, g12=> g33*g12,
            g21=> g33*g21, g22=> g33*g22,
            g31=> g33*g31, g32=> g33*g32,
            m1=> g33*m1, m2=> g33*m2,
            m3=> g33*m3 });

f0 = factor I1prime_0; f0 = f0#1#0;
f1 = factor I1prime_1; f1 = f1#1#0;
f2 = factor I1prime_2; f2 = f2#1#0;
f3 = factor I1prime_3; f3 = f3#1#0;
```

```
f4 = factor I1prime_4; f4 = f4#1#0;
f5 = factor I1prime_5; f5 = f5#1#0;
f6 = factor I1prime_6; f6 = f6#1#0;
f7 = factor I1prime_7; f7 = f7#1#0;
f8 = factor I1prime_8; f8 = f8#1#0;
f9 = factor I1prime_9; f9 = f9#0#0;
I1prime = ideal({f0,f1,f2,f3,f4,f5,f6,f7,
                 f8,f9});
I2prime  = eliminate({g33}, I1prime);

e = mingens I2prime
betti mingens I2prime
```

The output of this code are generators of the elimination ideal I_2', which is in this case generated by a set of five polynomials, each of which is of degree 5. For our proposed solvers from 4.5 point correspondences, $\mathbf{sH4.5_3}$ and $\mathbf{sH4.5_2}$, we used all of the 5 polynomials while for the solvers $\mathbf{sH5_3}$ and $\mathbf{sH5_2}$ we used the first polynomial e_1,

$$\begin{aligned} e_1 = {} & g_{12}^2 m_1\, m_2\, m_3 + g_{22}^2 m_1\, m_2\, m_3 + g_{32}^2 m_1\, m_2\, m_3 - \\ & g_{11}\, g_{12}\, m_2^2 m_3 - g_{21}\, g_{22}\, m_2^2 m_3 - g_{31}\, g_{32}\, m_2^2 m_3 - \\ & g_{11}\, g_{12}\, m_3^3 - g_{21}\, g_{22}\, m_3^3 - g_{31}\, g_{32}\, m_3^3 - g_{32}\, m_1\, m_2^2 + \\ & g_{31}\, m_2^3 + g_{32}\, m_1\, m_3^2 + g_{31}\, m_2\, m_3^2 - m_1\, m_2\, m_3 \ . \end{aligned} \tag{8}$$

We note that the stability of the solver would be similar irrespective of which one of these polynomials is used.

6. Synthetic Experiments

For our synthetic experiments, due to lack of space in the main paper, we only reported the plots of errors in the estimated rotations (Sec 3.1 in the main paper). The plots of errors in the estimated positions for calibrated solvers, as well as the plots of errors in the estimated focal lengths and positions for unknown focal length solvers, are provided here.

The relative error in the estimated position with respect to its ground truth value is computed as $t_\delta = \|\mathbf{t} - \mathbf{t_{GT}}\|/\|\mathbf{t_{GT}}\|$, and the relative error in the estimated focal length with respect to its ground truth value as $f_\delta = |f - f_{GT}|/|f_{GT}|$. Here, $\mathbf{t_{GT}}$ and f_{GT} respectively denote the ground truth value of the position and focal length. These errors are plotted in Figs. 2 and 3, which show the performance of the unknown focal length solvers and calibrated solvers in terms of numerical stability, solver performance in the presence of image noise, solver performance in the presence of close-to-planar scenes, and stability in the presence of forward motion and image noise. For both of the cases, we observe that our proposed solvers have comparable numerical stability to the state-of-the-art ones in estimating pose and focal length.

7. Details on the Localization Experiments

This section provides implementation details for the localization experiment from Sec. 3.3 of the main paper. In addition, we present a variation of the setup used in the main paper, where (a subset) of the top-20 retrieved images instead of the top-10 retrieved images is used.

All solvers (the variants of $\mathbf{sH5}$ and $\mathbf{sH4.5}$, $\mathtt{E_{5+1}}$, and $\mathtt{E_{4+2}}$) are integrated into the LO-RANSAC [8] implementation of [9]. We modified the implementation to skip the non-minimal solver inside the local optimization (LO) stage. We instead directly perform least-squares refinement of the estimated model using a random subset of the inliers of the best model found so far [8]. After LO-RANSAC, the estimated model is refined on all of its inliers using the same non-linear optimization approach. The non-linear optimization is implemented using the Ceres library [1].

For inlier counting and non-linear optimization, we consider a reprojection error: given a 2D-2D match and the poses of the generalized and the perspective cameras, we find the closest 3D points on the two lines corresponding to the viewing rays of the two matching image positions. The closest points are projected into the other image and we measure the mean of the two reprojection errors. For inlier counting, we use an inlier-outlier threshold of 5 pixels.

As mentioned in the paper, we randomly sample 5 correspondences from all available matches for our $\mathbf{sH5}$ and $\mathbf{sH4.5}$ solvers. We then select the variant that fits the sample, *e.g.*, $\mathbf{sH5}_2$ if at most two of the matches in the sample come from the same image in the generalized camera. In the case where all matches come from the same image, it is not possible to recover the translation scale. We thus simply discard the sample and continue with the next RANSAC iteration. However, this approach is not applicable to the $\mathtt{E_{5+1}}$, and $\mathtt{E_{4+2}}$ solvers since they only cover three possible configurations (5 matches from one camera and 1 from another; 4 matches from one camera and 2 from another; 4 matches from one camera and 1 each from two other cameras). Moreover, the chance that 4 or more out of the 6 matches in a sample come from the same camera is very small. Instead, we first randomly select two images for the $\mathtt{E_{5+1}}$ solver respectively three images for the $\mathtt{E_{4+2}}$ solver from the generalized camera. We then randomly select 4 / 5 matches from the first and 2 / 1 matches from the other images when using the $\mathtt{E_{4+2}}$ / $\mathtt{E_{5+1}}$ solver. Note that for the $\mathtt{E_{4+2}}$, the two additional images can be identical, *i.e.*, the additional two correspondences might come from the same camera. For a fair comparison of the tested solvers, we run RANSAC for a fixed number of iterations.

The solvers can return multiple solutions. Among these, we only use the solution with the smallest reprojection error on the minimal sample for inlier counting and discard the others. This approach significantly accelerates RANSAC and we did not observe a noticeable drop in performance for our solvers and the $\mathtt{E_{5+1}}$ solver.

Using the top-20 retrieved images. Tab. 2 in the main paper reports results on the Cambridge Landmarks dataset [4] when using the top-10 retrieved images. As an extension

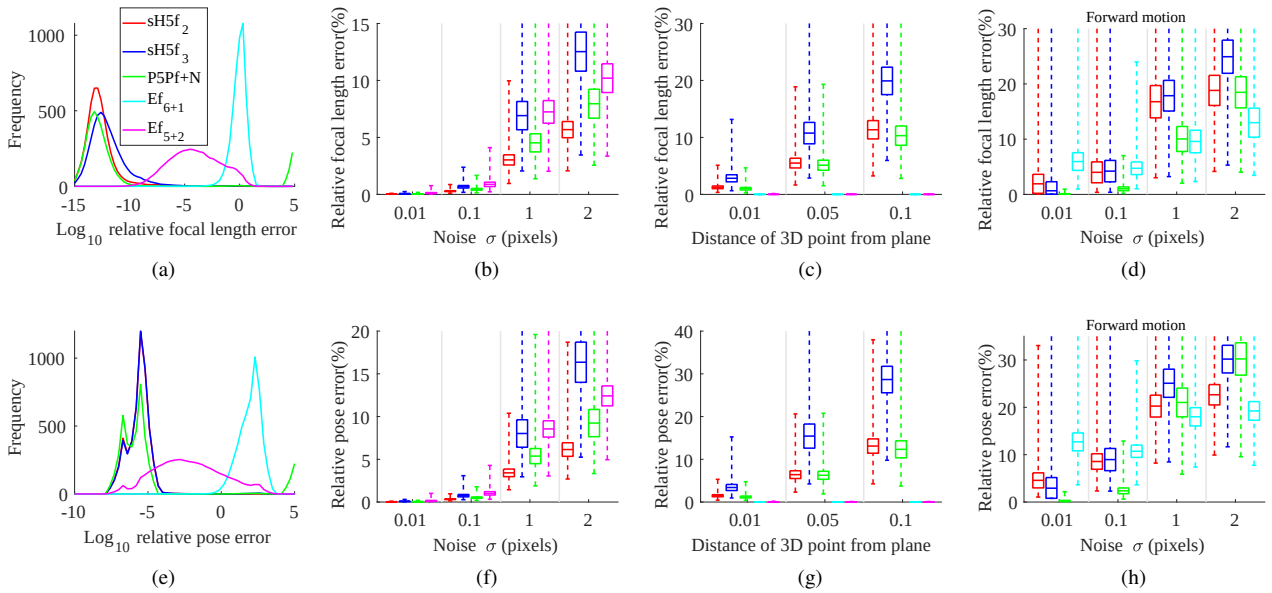

Figure 2. The focal length (top) and position (bottom) errors of solvers considering partially calibrated cameras. Plots (a,e) show the numerical stability, *i.e.*, histograms of errors, in the noise-free case. Plots (b,f) report the errors in the presence of image noise. Plots (c,g) show results on close-to-planar scenes. For (d,h), forward camera motion was considered.

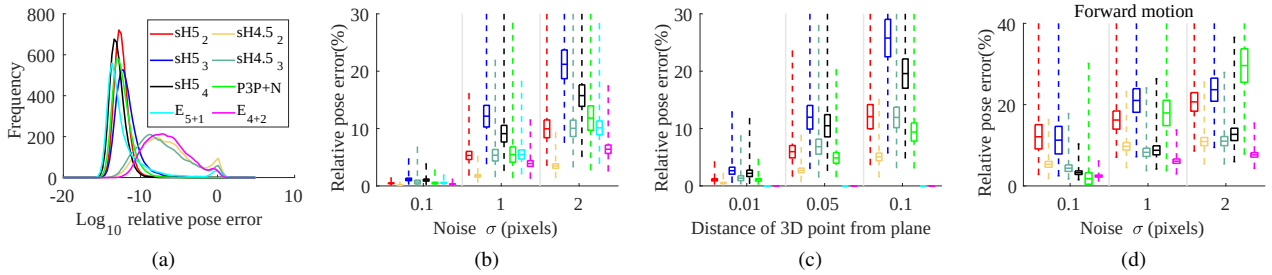

Figure 3. Pose error of solvers considering calibrated cameras. Plot (a) shows the numerical stability, *i.e.*, histogram of errors in the noise-free case. Plot (b) reports the errors in the presence of image noise. Plot (c) shows results on close-to-planar scenes. For (d), forward camera motion was considered.

to this experiment, we also use the top-20 retrieved images. We further experiment with sparsifying the set of retrieved images: [7, 11] first compute the essential matrices between the perspective image and the images in the generalized camera. They then use these relative poses to triangulate the pose of the perspective image. Naturally, larger baselines between the images in the generalized camera lead to more stable triangulations. Thus, [7, 11] report that better results are obtained when ensuring that the images used to define the generalized camera have certain baselines. To analyze whether this also holds for the structure-less localization methods used in this paper, we employ a simple approach: in the order in which the images were retrieved, we first select the top-retrieved image. The top-i-th retrieved image is then selected if it is at least X meters away from all previously selected images. We experiment with thresholds of $X = 1$ meters and $X = 2$ meters.

Tab. 1 shows the results of this experiment. Note that we are excluding the $\mathtt{E_{4+2}}$ solver as Tab. 2 in the main paper shows that it performs worse than the other solvers in terms of pose accuracy and run-times. For completeness, we reprint the part of Tab. 2 corresponding to 1,000 RANSAC iterations. As can be seen, using the top-20 instead of top-10 retrieved images to define the generalized cameras in general reduces the rotation errors without much impact on the position errors. Selecting a subset of the top-20 retrieved images based on the distance threshold X does not result in better pose estimates than using all top-20 images. However, it leads to faster run-times as the generalized cameras contain fewer images (typically fewer than 10), resulting in fewer matches. In addition, the results for subsets of the top-20 retrieved images show that smaller rotation errors and similar position errors can be achieved using fewer than 10 images compared to simply using the

Method	King's College			Old Hospital			Shop Facade			St. Mary Church			Avrg. all	
	pos.	rot.	time	pos.	rot.	time	pos.	rot.	time	pos.	rot.	time	pos.	rot.
top-10 retrieved images														
$\mathtt{E_{5+1}}$ [10] (1k iter.)	**0.19** / 0.33	**0.34** / **0.48**	0.82	**0.42** / **1.10**	**0.75** / **1.78**	0.39	**0.06** / **0.10**	**0.29** / **0.44**	0.37	**0.11** / **0.15**	**0.38** / **0.55**	0.49	**0.20**	**0.44**
$\mathtt{E_{4+2}}$ [10] (1k iter.)	**0.20** / 0.42	0.35 / 0.60	0.75	0.83 / 2.51	1.55 / 3.88	0.56	**0.07** / **0.16**	**0.32** / 0.70	0.53	0.20 / 0.70	0.71 / 2.17	0.59	0.33	0.73
ours (sH5) (1k iter.)	**0.20** / **0.31**	**0.34** / **0.48**	**0.33**	0.46 / **1.03**	0.89 / 2.47	**0.16**	**0.06** / **0.10**	**0.29** / **0.45**	**0.16**	0.13 / 0.43	0.47 / 1.35	**0.20**	**0.21**	**0.50**
ours (sH4.5) (1k iter.)	**0.19** / **0.30**	**0.33** / **0.46**	**0.52**	**0.40** / 1.21	**0.74** / **1.91**	**0.27**	**0.06** / **0.10**	**0.29** / **0.44**	**0.26**	**0.12** / **0.17**	**0.40** / **0.59**	**0.33**	**0.20**	**0.44**
top-20 retrieved images														
$\mathtt{E_{5+1}}$ [10] (1k iter.)	**0.17** / **0.26**	**0.32** / **0.40**	1.43	**0.41** / **1.05**	**0.80** / **1.69**	0.58	**0.06** / **0.09**	**0.26** / **0.41**	0.52	**0.10** / **0.14**	**0.38** / **0.51**	0.79	**0.19**	**0.44**
ours (sH5) (1k iter.)	**0.17** / **0.27**	0.33 / **0.41**	**0.58**	0.46 / 1.27	0.91 / 2.31	**0.23**	**0.06** / **0.10**	**0.26** / **0.45**	**0.20**	0.12 / 0.29	**0.44** / 0.84	**0.32**	0.20	0.49
ours (sH4.5) (1k iter.)	**0.16** / **0.26**	**0.29** / **0.41**	**0.89**	**0.33** / **0.73**	**0.65** / **1.25**	**0.40**	**0.05** / **0.09**	**0.25** / **0.41**	**0.33**	**0.11** / **0.15**	**0.38** / **0.53**	**0.50**	**0.16**	**0.39**
top-20 retrieved images, 1 meter minimal distance between selected retrieved images														
$\mathtt{E_{5+1}}$ [10] (1k iter.)	**0.18** / **0.27**	**0.33** / **0.42**	0.75	**0.35** / 0.85	0.74 / **1.37**	0.36	**0.06** / **0.09**	**0.28** / **0.41**	0.27	**0.11** / **0.14**	**0.40** / **0.53**	0.43	**0.18**	**0.44**
ours (sH5) (1k iter.)	**0.18** / **0.26**	**0.32** / **0.42**	**0.31**	**0.37** / **0.84**	**0.71** / **1.33**	**0.14**	**0.07** / **0.10**	**0.28** / **0.46**	**0.12**	**0.14** / 0.24	**0.50** / 0.78	**0.17**	**0.19**	0.45
ours (sH4.5) (1k iter.)	**0.19** / 0.28	0.35 / **0.43**	**0.49**	0.38 / **0.81**	**0.67** / **1.37**	**0.25**	**0.06** / **0.09**	**0.27** / **0.41**	**0.19**	**0.11** / **0.16**	**0.40** / **0.56**	**0.29**	**0.19**	**0.42**
top-20 retrieved images, 2 meter minimal distance between selected retrieved images														
$\mathtt{E_{5+1}}$ [10] (1k iter.)	**0.18** / **0.28**	**0.33** / **0.43**	0.50	**0.40** / **0.80**	**0.79** / **1.32**	0.22	**0.07** / **0.10**	**0.31** / **0.45**	0.19	**0.11** / **0.15**	**0.37** / **0.54**	0.30	**0.19**	**0.45**
ours (sH5) (1k iter.)	**0.19** / **0.29**	0.35 / **0.46**	**0.20**	0.44 / 0.95	0.83 / 1.74	**0.10**	**0.08** / **0.12**	**0.31** / 0.53	**0.09**	0.15 / 0.57	0.50 / 1.61	**0.12**	**0.22**	0.50
ours (sH4.5) (1k iter.)	**0.19** / **0.28**	**0.34** / **0.43**	**0.33**	**0.39** / **0.64**	**0.69** / **1.14**	**0.18**	**0.07** / **0.10**	**0.31** / **0.47**	**0.16**	**0.12** / **0.16**	**0.41** / **0.58**	**0.22**	**0.19**	**0.44**

Table 1. Localization results on Cambridge Landmarks [4]. We report the median/mean position (in meters) and rotation (in degrees) errors, and the mean RANSAC time (in seconds). We also report the average median position and rotation error over all four scenes. We show results obtained with the top-10 retrieved images (identical to the results in Tab. 2 of the main paper), the top-20 retrieved images, and subsets of the top-20 retrieved images where there is at least 1 / 2 meter distance between the selected retrieved images (see Sec. 7 for details). We show results for fixing the number of RANSAC iterations to 1000. Best and second best results are shown in red and blue. We do not show results for the $\mathtt{E_{4+2}}$ solver for the top-20 retrieved images as it is both slower and less accurate compared to the other solvers.

top-10 retrieved images.

Consistent with the results for the top-10 retrieved images, our sH4.5 solvers perform similar as the $\mathtt{E_{5+1}}$ solver on average over all scenes while offering faster run-times.

References

[1] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. http://ceres-solver.org. 6

[2] Daniel R. Grayson and Michael E. Stillman. Macaulay2, a software system for research in algebraic geometry. Available at http://www.math.uiuc.edu/Macaulay2/. 4

[3] D. Hook and P. R. McAree. Using sturm sequences to bracket real roots of polynomial equations. *Graphics gems*, pages 416–422, 1990. 3, 4

[4] Alex Kendall, Matthew Grimes, and Roberto Cipolla. PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. In *ICCV*, 2015. 6, 8

[5] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Automatic generator of minimal problem solvers. In *European Conference on Computer Vision*, pages 302–315. Springer, 2008. 1

[6] Viktor Larsson, Kalle Åström, and Magnus Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *CVPR*, volume 2, page 4, 2017. 1, 2, 4

[7] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala. Camera Relocalization by Computing Pairwise Relative Poses Using Convolutional Neural Network. In *ICCV Workshops*, 2017. 7

[8] Karel Lebeda, Jirı Matas, and Ondrej Chum. Fixing the locally optimized ransac–full experimental evaluation. In *British machine vision conference*, pages 1–11. Citeseer, 2012. 6

[9] Torsten Sattler et al. RansacLib - A Template-based *SAC Implementation, 2019. 6

[10] Enliang Zheng and Changchang Wu. Structure from motion using structure-less resection. In *International Conference on Computer Vision (ICCV)*, 2015. 8

[11] Qunjie Zhou, Torsten Sattler, Marc Pollefeys, and Laura Leal-Taixe. To Learn or Not to Learn: Visual Localization from Essential Matrices. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019. 7