Finding the exact rotation between two images independently of the translation

Laurent Kneip[1], Roland Siegwart[1], and Marc Pollefeys[2]

[1] Autonomous Systems Lab, ETH Zurich
[2] Computer Vision and Geometry Group, ETH Zurich

Abstract. In this paper, we present a new epipolar constraint for computing the rotation between two images independently of the translation. Against the common belief in the field of geometric vision that it is not possible to find one independently of the other, we show how this can be achieved by relatively simple two-view constraints. We use the fact that translation and rotation cause fundamentally different flow fields on the unit sphere centered around the camera. This allows to establish independent constraints on translation and rotation, and the latter is solved using the Gröbner basis method. The rotation computation is completed by a solution to the cheiriality problem that depends neither on translation, nor on feature triangulations. Notably, we show for the first time how the constraint on the rotation has the advantage of remaining exact even in the case of translations converging to zero. We use this fact in order to remove the error caused by model selection via a non-linear optimization of rotation hypotheses. We show that our method operates in real-time and compare it to a standard existing approach in terms of both speed and accuracy.

1 Introduction

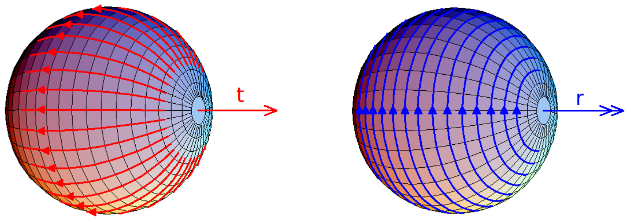
Numerous works over the past decade proved the feasibility of doing robust structure-from-motion with a single camera only. Some important examples are given by Nistér et al. [1], Davison et al. [2] and Klein and Murray [3]. The biggest problem from a geometric point of view has been identified to be the robust estimation of the rotation between successive camera frames. The effect of certain small translations and rotations on the displacement of the features in the image plane can be very similar and the problem of disambiguation is even amplified if the distribution of the features in the image plane is too inhomogenous.

Klein and Murray [3] find the rotation between two frames by warping and rotating *small blurry images* of the camera frames and then minimizing the sum of squared differences in between them. Another approach has recently been presented by Kneip et al. [4], who solve this problem by taking short-term integrals of gyroscopic signals of an additional inertial measurement unit into account. However, both approaches only deliver an approximate value for the rotation between two frames. In this paper, we present an exact and robust geometric formulation for an independent computation of the rotation between two camera frames.

The subject of this paper is clearly related to one of the most fundamental and traditional problems in geometric vision, namely the determination of the relative pose between two images. The input information is given in form of point correspondences between the two images, or—in the calibrated case—unit vectors pointing from each viewpoint to the jointly observed 3D points. The goal is to find the translation and rotation between the two frames. Of high interest are solutions that use a minimum number of points and can thus be employed in robust hypothesize-and-test schemes such as the RANSAC approach [5]. The most important minimal solutions are the 5-point solvers by Nistér [6], Stewénius et al. [7], and Kukelova et al. [8] and the famous 8-point solver by Longuet-Higgins [9]. The first solution was presented in 1913 by Kruppa [10]. The main problem with all these approaches—apart from potentially degenerate structure configurations and multiplicity of solutions—is that they mix the parameters for translation and rotation and thus become numerically intractable in the case of vanishing translation magnitudes. This issue notably applies to any parametrization that involves the fundamental or essential matrix between the two images. The way this problem has been handled in the literature consists of applying a model selection criterium that decides whether the displacement has too small translation or not and then solving for pure rotation in case it does. This has for instance been presented by Torr et al. [11]. While this approach achieves robust behavior, it has the downside of not delivering exact rotations in the case of sufficiently small translations. Moreoever, it is impossible to reduce this error in a two-view batch optimization. Bundle adjustment depends on feature triangulations, which are undefined if the parallax is neglected and simply set to zero.

In 2009, Kalantari et al. [12] presented a new direct parametrization in terms of translation and rotation. Their method results in a system of 6 equations in 6 unknowns and a Gröbner basis with 40 base monomials. They claim that their approach stays robust in the case of zero translation, however without presenting any quantitative results in this direction. They also solve the Gröbner basis online, which is not the most efficient way regarding the fact that the path of the algorithm actually remains constant for one specific problem. In 2010, Lim et al. [13] present a solution for decoupled translation and rotation computation. However, their method imposes constraints on the distribution of the features on the unit sphere and hence does not work in the general case.

In this paper, we present new general epipolar constraints that allow the independent computation of rotation and translation between two frames based on the individual characteristics of the resulting optical flow on the unit sphere. We obtain a system of equations that allows to compute the rotation between two frames generally and independently of the translation using a minimum number of points. Notably, we also show how the rotation constraint stays numerically robust even in the case of vanishing translation magnitudes and hence allows for an exact computation of the rotation in any case. Section 2 shows the



(a) (b)

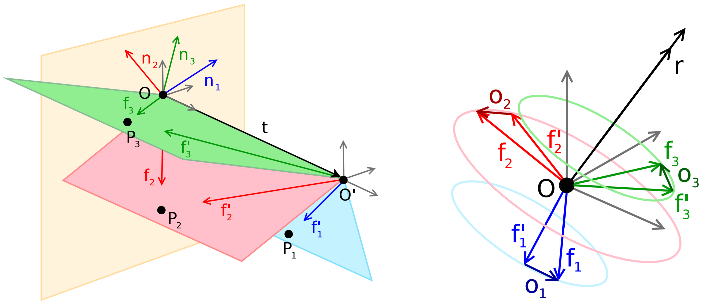
Fig. 1. Optical flow of features caused by camera translation (a) and rotation (b).

derivation of computationally independent constraints on translation and rotation. Section 3 illustrates how the rotation can be solved using the Gröbner basis method. Section 4 presents a non-linear refinement of the solutions and notably how our constraint allows for an exact computation of the rotation for any given translation. Section 5 then presents a concise evaluation and comparison of our algorithm in terms of accuracy, noise resilience and execution time.

2 Translation independent rotation constraint

2.1 Independence of rotation and translation computation

A good illustration of why the rotation and translation of a camera displacement can be separated is given by the optical flow on the unit sphere. As shown in Figure 1, the optical flow caused by a translation consists of the shortest field lines emerging from/ending at the intersection points between the sphere and the translation direction. The rotational optical flow is fundamentally different and consists of field lines contained in parallel planes of which the rotation vector is a normal vector.



(a) (b)

Fig. 2. Geometric relationships of unit feature observation vectors in the case of pure translation (a) and pure rotation (b).

To the end of obtaining implicit constraints on rotation and translation, we introduce the unit feature vectors $\mathbf{f}_i$ and $\mathbf{f'}_i$ describing the feature observations from viewpoint 1 and 2, respectively. Figure 2 shows the influence of a pure translation or a pure rotation on the feature observations. In the case of a pure translation, each correspondence $(\mathbf{f}_i, \mathbf{f'}_i)$ spans a so-called epipolar plane. Moreover, all of the epipolar planes intersect in the line defined by the translation vector. In other words, the normal vectors of the epipolar planes all need to be contained in the normal plane of the translation vector. A normal vector of each epipolar plane is in this special case easily given by $\mathbf{n}_i = \mathbf{f}_i \times \mathbf{f'}_i$. The constraint for a purely translational optical flow thus translates to all cross-products $\mathbf{f}_i \times \mathbf{f'}_i$ being coplanar. In the case of a pure rotation, the situation is even easier. Since the field lines are all contained in parallel planes, it immediately follows that the constraint for a pure rotation translates to all optical flow vectors $\mathbf{o}_i = \mathbf{f}_i - \mathbf{f'}_i$ being coplanar as well.

The idea for independent rotation and translation computation finally reads as follows:

- *rotation* is correctly compensated if and only if the feature correspondences $(\mathbf{f}_i, c_{\mathbf{R}}(\mathbf{f'}_i))$ fullfill the properties of a *purely translational optical flow*, and
- *translation* is correctly compensated if and only if the feature correspondences $(\mathbf{f}_i, c_{\mathbf{t}}(\mathbf{f'}_i))$ fullfill the properties of a *purely rotational optical flow*,

where $c_{\mathbf{R}}(.)$ and $c_{\mathbf{t}}(.)$ are the functions that compensate for rotation and translation, respectively. If $\mathbf{R}$ denotes the rotation from viewpoint 2 to viewpoint 1, the rotation-compensated observations from viewpoint 2 become $c_{\mathbf{R}}(\mathbf{f'}_i) = \mathbf{R}\mathbf{f'}_i$. In this case we need to have a purely translational optical flow and thus all vectors $\mathbf{f}_i \times \mathbf{R}\mathbf{f'}_i$ need to be coplanar. The minimum number of vectors for expressing coplanarity is three and a simple way to encode coplanarity/linear dependency is given by the determinant of these three vectors being zero. A translation independent constraint on the rotation is finally given by

$$|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f'}_1) \ \ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f'}_2) \ \ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f'}_3)| = 0. \tag{1}$$

If $\mathbf{t'}$ denotes the translation from viewpoint 2 to viewpoint 1 seen from viewpoint 2, the translation compensated observations from viewpoint 2 become $c_{\mathbf{t'}}(\mathbf{f'}_i) = \frac{\eta'_i \mathbf{f'}_i - \mathbf{t'}}{||\eta'_i \mathbf{f'}_i - \mathbf{t'}||}$, with η'_i denoting the depths of the features seen from viewpoint 2. This time we need to have a purely rotational optical flow, meaning that all vectors $\mathbf{f}_i - \frac{\eta'_i \mathbf{f'}_i - \mathbf{t'}}{||\eta'_i \mathbf{f'}_i - \mathbf{t'}||}$ need to be coplanar. Following a similar argumentation to the one previously developped, a rotation independent constraint on the translation is given by

$$\left|(\mathbf{f}_1 - \frac{\eta'_1 \mathbf{f'}_1 - \mathbf{t'}}{||\eta'_1 \mathbf{f'}_1 - \mathbf{t'}||}) \ \ (\mathbf{f}_2 - \frac{\eta'_2 \mathbf{f'}_2 - \mathbf{t'}}{||\eta'_2 \mathbf{f'}_2 - \mathbf{t'}||}) \ \ (\mathbf{f}_3 - \frac{\eta'_3 \mathbf{f'}_3 - \mathbf{t'}}{||\eta'_3 \mathbf{f'}_3 - \mathbf{t'}||})\right| = 0 \tag{2}$$

It is interesting to see that the depths of the features only appear in the translation constraint. The independent constraint on the rotation (1) appears fairly compact. It is the main subject of the remaining of the paper and we name it the *epipolar plane normal coplanarity constraint.*

2.2 Constraining the rotation

A rotation encodes 3 degrees of freedom. We thus need at least three epipolar plane normal coplanarity constraints in order to fully constrain the rotation. Using two additional features $\mathbf{f}_4$ and $\mathbf{f}_5$, we obtain the following system of equations to calculate the rotation

$$\begin{cases} |(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}'_1)\ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}'_2)\ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f}'_3)| = 0 \\ |(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}'_1)\ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}'_2)\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}'_4)| = 0\ . \\ |(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}'_1)\ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}'_2)\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}'_5)| = 0 \end{cases} \tag{3}$$

In addition to these equations, there might be—depending on the employed parametrization—additional constraints enforcing the rotation matrix to actually be a rotation matrix. As illustrated in the following sections, the choice of the parametrization has a huge impact on the complexity of the solution.

2.3 The case of zero translation

The constraints on the rotation (3) turn out to be still valid even if the translation is zero. For a correct rotation compensation, the entities $\mathbf{f}_i \times \mathbf{R}\mathbf{f'}_i$ then ideally become zero, which does however not change the fact that the determinant constraint is zero in this case. Notably, since the mentioned entities deviate from zero if a wrong rotation matrix is chosen—even in the case of pure rotation—, the determinant is zero if and only if the rotation is correctly compensated. The only exception is given with degenerate feature configurations. For instance, if all the features are situated along the equator, we would have an optical flow that could be explained by both our rotation-only and translation-only constraints. A similar development than above can be done for the translation constraint.

3 5-point minimal solution

3.1 Finding a Gröbner basis

Despite the compact look, equation system (3) is arduous to solve. It is a multivariate polynomial equation system commonly solved via the Gröbner basis method. A good introduction to the approach can be found in [14]. The method consists of defining a monomial ordering over the polynomial terms and then iteratively generating and reducing new polynomials inside the ideal (the so-called S-polynomials) until a set of polynomials with certain desired criteria w.r.t. solvability is obtained. This method has been applied to numerous minimal problems in geometric vision, among which the most well known approaches certainly are represented by the classical 5-point essential matrix solutions of Nistér [6] and Stewénius [7]. [3] As a matter of fact, the path followed by the algorithm remains

[3] The solution of Nistér actually turns out to be equivalent to computing a lexicographical Gröbner basis instead of a graded-reverse lexicographical Gröbner basis as done by Stewénius. It results in a univariate 10th degree polynomial, which can be solved substantially faster than a 10x10 action matrix.

constant for one specific problem and needs to be resolved only once exactly, which means outside the algebraic field of real numbers. Hence the way such polynomial systems are solved is by chosing random coefficients in a prime field and then doing the computation therein. The path of the solution can be *traced* offline in the prime field and afterwards applied online to the effective feature-depending coefficients. The final algorithm is fast since we do no longer need to check for all polynomial reductions, but directly generate and reduce only the necessary S-polynomials.

The computation of a Gröbner basis can be extremely long and the complexity depends to a large extend on the initial parametrization of the problem. It is influenced by the following factors: 1) the order of the equations; 2) the number of equations; 3) the number of unknowns; and 4) the chosen monomial ordering. Using our custom-made Gröbner basis computation software (∼6000 lines of code), we tried out many different rotation matrix parametrizations as for instance Cayley [15], quaternion, and Thompson [16] rotations. However, due to the increasing order of the equations, our conclusion is that this system is best solved using the standard rotation matrix parametrization $\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$. Moreover, it is best to increase the number of equations as much as possible and thus take into account as many epipolar plane normal coplanarity constraints as possible. Finally, it is also best to formalize the constraints on the rotation matrix in terms of multiple quadratic constraints and leave out the third order determinant constraint.

$$\begin{cases}
|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}_1')\ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}_2')\ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f}_3')| = 0 \\
|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}_1')\ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}_2')\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}_4')| = 0 \\
|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}_1')\ (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}_2')\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}_5')| = 0 \\
|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}_1')\ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f}_3')\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}_4')| = 0 \\
|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}_1')\ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f}_3')\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}_5')| = 0 \\
|(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}_1')\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}_4')\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}_5')| = 0 \\
|(\mathbf{f}_2 \times \mathbf{R}\mathbf{f}_2')\ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f}_3')\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}_4')| = 0 \\
|(\mathbf{f}_2 \times \mathbf{R}\mathbf{f}_2')\ (\mathbf{f}_3 \times \mathbf{R}\mathbf{f}_3')\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}_5')| = 0 \\
|(\mathbf{f}_2 \times \mathbf{R}\mathbf{f}_2')\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}_4')\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}_5')| = 0 \\
|(\mathbf{f}_3 \times \mathbf{R}\mathbf{f}_3')\ (\mathbf{f}_4 \times \mathbf{R}\mathbf{f}_4')\ (\mathbf{f}_5 \times \mathbf{R}\mathbf{f}_5')| = 0 \\
r_{11}^2 + r_{12}^2 + r_{13}^2 = 1 \\
r_{11}r_{21} + r_{12}r_{22} + r_{13}r_{23} = 0 \\
r_{11}r_{31} + r_{12}r_{32} + r_{13}r_{33} = 0 \\
r_{21}^2 + r_{22}^2 + r_{23}^2 = 1 \\
r_{21}r_{31} + r_{22}r_{32} + r_{23}r_{33} = 0 \\
r_{31}^2 + r_{32}^2 + r_{33}^2 = 1 \\
r_{21}r_{32} - r_{22}r_{31} - r_{13} = 0 \\
r_{12}r_{31} - r_{11}r_{32} - r_{23} = 0 \\
r_{11}r_{22} - r_{12}r_{21} - r_{33} = 0 \\
r_{22}r_{33} - r_{23}r_{32} - r_{11} = 0 \\
r_{13}r_{32} - r_{12}r_{33} - r_{21} = 0 \\
r_{12}r_{23} - r_{13}r_{22} - r_{31} = 0 \\
r_{11}^2 + r_{21}^2 + r_{31}^2 = 1 \\
r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} = 0 \\
r_{11}r_{13} + r_{21}r_{23} + r_{31}r_{33} = 0 \\
r_{12}^2 + r_{22}^2 + r_{32}^2 = 1 \\
r_{12}r_{13} + r_{22}r_{23} + r_{32}r_{33} = 0 \\
r_{13}r_{21} - r_{11}r_{23} - r_{32} = 0 \\
r_{23}r_{31} - r_{21}r_{33} - r_{12} = 0 \\
r_{11}r_{33} - r_{13}r_{31} - r_{22} = 0
\end{cases} \tag{4}$$

The final parametrization is given with equation system (4). It consists of 10 cubic coplanarity constraints and 20 quadratic constraints on the rotation matrix and is solvable—using the grevlex monomial ordering and $r_{33} > r_{32} > r_{31} > r_{23} > r_{22} > r_{21} > r_{13} > r_{12} > r_{11}$—via 36 S-polynomial reductions only. We finally obtain a Gröbner basis in the 20 base monomials r_{12}^2, $r_{11}r_{33}$, $r_{11}r_{32}$, $r_{11}r_{31}$, $r_{11}r_{23}$, $r_{11}r_{22}$, $r_{11}r_{21}$, $r_{11}r_{13}$, $r_{11}r_{12}$, r_{11}^2, r_{33}, r_{32}, r_{31}, r_{23}, r_{22}, r_{21}, r_{13}, r_{12}, r_{11}, 1. The SVD of the action matrix then finally leads to 20 direct solutions for the rotation matrix, which is reasonable since each essential matrix actually represents two possible rotation matrices after decomposition. Our Gröbner basis tool basically combines the results of Buchberger [17], Gebauer and Möller [18], and Giovini et al. [19].

The code we extract from the trace (∼8000 lines) operates on a large matrix in a similar way to Faugère's F4 [20] or the code extracted from Kukelova's framework [21], however substantially faster since operating on a much smaller matrix. This is achieved by the following modifications:

- The number of columns is reduced by only representing the monomials that effectively appear along the computation.
- When creating new S-polynomials, no copies of the pair of generators multiplied by the subsidiary monomials to the LCM are instantiated.
- When creating new S-polynomials, the canceling leading term is left out immediately.
- When reducing S-polynomials, no copies of the reductors multiplied by the quotient term are instantiated.
- We recursively backtrace the S-polynomials that are effectively necessary for the final basis and generate only those.

The operation is supported by vector-functions that access the right generator and reductor coefficients for combining terms without actually multiplying the polynomial by the subsidiary monomials to the LCM (Least Common Multiple). Our matrix is 66 × 197 and the Gröbner basis is computed in less than 0.3ms.

3.2 Selection of the right solution

The 20 solutions of the action matrix SVD exist in the complex domain and only those with a sufficiently low imaginary part are interesting. Wrong solutions are easily rejected by thresholding the magnitude of the imaginary parts of the singular values from the SVD. The next consistency check then consists in verifying if the determinants of the rotation matrices equal to one, which again allows to purge wrong solutions. Moreover, we back-substitute the remaining rotation matrices into the original constraints and put a threshold on the resulting error. Another way of identifying wrong solutions is given by solving the cheiriality ambiguity, as we will explain next.

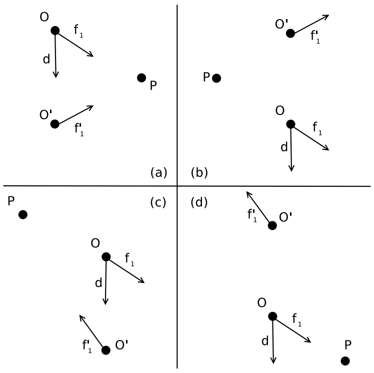


Fig. 3. Ambiguity for translation and rotation known as cheiriality.

From the essential matrix decomposition, we know that the obtained rotation matrices may appear in pairs of reflected matrices. The same accounts for the translations. Figure 3 illustrates how this finally leads to 4 possible $(\mathbf{R}, \mathbf{t})$-pairs for each feature triangulation. The classical way of solving this problem is to triangulate features and then check whether they lie in front or behind the camera. Since we only have rotation matrices, we focus on a constraint that allows us to effectuate the disambiguation without actually deriving the translation or doing feature-triangulation. First of all, it is important to notice that the criterium whether a certain point lies in front or behind the camera plane is not very accurate. For an omnidirectional camera for instance, the front and the back of a camera does not tell anything about the visibility of the feature. We therefore change this criterium to something more meaningful, namely check whether the

3D point lying on a line defined by the feature vector also lies on the *beam* defined by the feature vector. Secondly, looking at Figure 3, we notice that (a) and (b) have the same rotation, and (c) and (d) as well. The difference between both cases only consists in the sign of the translation. For (a) and (b), the 3D point is lying either on both or none of the beams. For (c) and (d), the 3D point is lying on one of the beams only. **d** represents a direction vector and therefore it is equal to either plus or minus the translation vector. The direction vector can be obtained by any cross-product of two epipolar plane normal vectors and thus we have

$$\mathbf{d} = \pm\mathbf{t} = \mathbf{n}_1 \times \mathbf{n}_2 = (\mathbf{f}_1 \times \mathbf{R}\mathbf{f}'_1) \times (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}'_2). \tag{5}$$

For the correct rotation matrix (cases (a) and (b)), we observe that the cross-product between the direction vector **d** and $\mathbf{f}_1$ needs to point to the same direction than the cross-product between the direction vector **d** and $\mathbf{R}\mathbf{f}'_1$. In other words, the dot-product of both cross-products must be positive. This translates into the following inequality constraint on each rotation matrix

$$(\mathbf{d} \times \mathbf{f}_1) \cdot (\mathbf{d} \times \mathbf{R}\mathbf{f}'_1) > 0$$

$$\Rightarrow \{(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}'_1) \times (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}'_2) \times \mathbf{f}_1\} \cdot \{(\mathbf{f}_1 \times \mathbf{R}\mathbf{f}'_1) \times (\mathbf{f}_2 \times \mathbf{R}\mathbf{f}'_2) \times \mathbf{R}\mathbf{f}'_1\} > 0 \tag{6}$$

We finally found a constraint that allows the disambiguation of rotation matrices without having to derive the translation or 3D point information. Note that any pair of features can be used for the disambiguation.

3.3 The case of zero translation

Despite the fact that our original rotation constraint is still valid for zero translation, the computation still deteriorates when the translation converges to zero. This phenomenon can be explained as follows. When the translation approaches zero, the 20 solutions of the 5-point algorithm converge to a single solution, which is natural since the rotation in the case of zero translation is uniquely defined by two features already. The side-effect is that—when following the fixed trace of the Gröbner basis computation—we obtain similar polynomials and hence the reductions result in almost zero coefficients in the middle of the computation. The solution parametrization having too high complexity for this specific case, the path finally becomes numerically unstable.

We handle this problem similar to Torr et al. [11] and perform model selection. We try to find a suitable rotation using two features only and accept it in case the unrotated features from view-point 2 turn out to be close enough to the features from view-point 1. Accepting the origin as a third virtual point and following the point-set alignment approach presented in [22], this can easily be done via computing the SVD

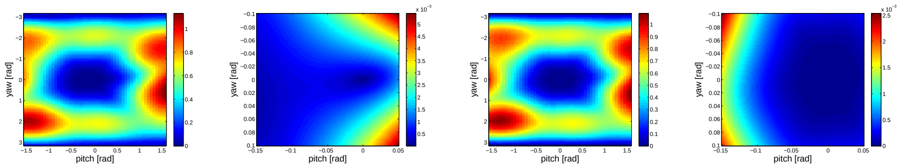
$$\mathbf{U} \cdot \mathbf{D} \cdot \mathbf{V}^* = SVD(\sum_{i=1}^{3}(\mathbf{f}'_i - \bar{\mathbf{f}}') \cdot (\mathbf{f}_i - \bar{\mathbf{f}})^t)) \Rightarrow \mathbf{R} = \mathbf{U} \cdot \mathbf{V}^t, \tag{7}$$

and then checking the magnitude of the differences $\mathbf{f}_i - \mathbf{R}\mathbf{f'}_i$. The drawback of this approach is that there remains a small error in the computation of the rotation for small translations. Besides, since the 20 local minima are very close to each other, this error can not be corrected via a non-linear optimization with 5 features. Even a two-view batch optimization would be infeasible since depending on feature triangulations, which are undefined if neglecting the parallax and setting it to zero.

4 Non-linear refinement

The final approach to remove the error caused by model selection and find a unique and exact rotation consists of taking more features into account and performing non-linear optimization over all resulting epipolar plane normal coplanarity constraints. In this paper, we will focus on the minimal variant of taking only one additional feature into account, although the accuracy of the optimization can obviously be improved by increasing this number. The minimization can in principle be extended to any number of features. The number of resulting constraints for 6 points is given by the combination $\binom{6}{3} = \frac{6!}{(6-3)!\cdot 3!} = 20$. The minimization is carried out over the Cayley parameters of a rotation matrix [15]. While this parametrization was unsuited for the Gröbner basis computation, it turns out to provide good properties for an iterative optimization scheme, namely symmetric parametrization in function of only 3 parameters and absence of additional constraints on the rotation matrix.

As illustrated in Figure 4(a), the norm of the errors of the constraints behaves non-linearly. Moreover, as shown in Figure 4(b)—a closeup of the region around the global minimum—, the manifold is not free of local minima. However, it equals to zero for the correct rotation values only (in this case $(0\ 0\ 0)^t$). The figure only shows the error over pitch and yaw angle variations, but the behavior for roll angle variations is comparable. The reason for the displayed local minimum is actually given by rotations that attempt to explain the translation as well (in the illustrated example we have a translation along x and the effect



(a) (b) (c) (d)

Fig. 4. (a): Norm of the errors of all epipolar plane normal coplanarity constraints for 6 features and pitch and yaw deviations. (b): Closeup of the region around the global minimum. (c): Norm of the error for zero translations. (d): Same closeup of the region around the global minimum for zero translation.

is close to the one of a pitch angle variation (rotation around y)). The main advantage of our constraint is shown in Figure 4(c): the cost-function remains robust for zero translation and thus paves the way for computing exact rotations even in the case of translations down to zero. Moreover, Figure 4(d) shows that in the event of a pure rotation, the mentioned local minimum even disappears. On the other hand, this means that for very small translations, we have local minima very close to the absolute minimum. A good minimization is therefore only possible by taking additional measures not to end up in a local minimum.

Taking the remaining rotations from the 5-point algorithm (or the unique rotation from two features in case of a small translation) as initial points for the optimization, we can finally identify a unique and exact rotation between the two view-points with only 6 points. The entire method can be summarized as follows:

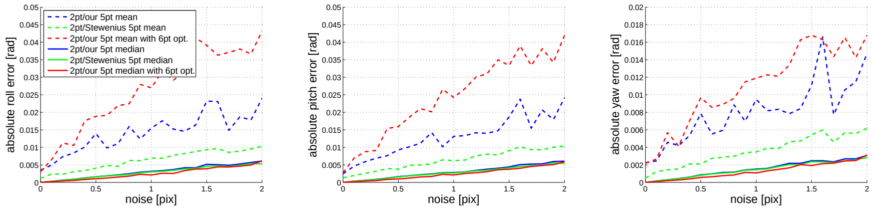
- Select 2 of the 6 features and try to compute the rotation as described in Section 3.3
  - *If* the translation is small enough, retain this unique hypothesis
  - *Else*, compute the rotation hypotheses using the 5-point Gröbner basis method and reject as many as possible as described in Section 3.2
- Finally, optimize all found hypotheses using nonlinear refinement over all 6 points and select the solution with least squares error.

5 Results

The algorithm we present is thoroughly tested by means of synthetic data and compared to Stewénius' [7] 5-point solution for computing the rotation and translation between two images. This is a clean, accurate and popular approach to epipolar geometry computation. The synthetic data consist of 1'000 3D points that are evenly distributed below the fixed downlooking camera at viewpoint 1, with depths varying uniformly between 4 and 8 in the vertical direction. For each test, we then select a randomly displaced viewpoint 2. The magnitude of the translation is uniformly distributed between 0 and 2 in order to ensure that the position of viewpoint 2 stays within a reasonable distance of the structure. For each experimental run, synthetic 2D-2D correspondences are then created by randomly selecting 6 points from the entire point set and reprojecting them on the unit sphere of our two viewpoints. Depending on the experiment, a different level of white Gaussian noise ranging from 0 to 2 pixels is then added to the measurements by assuming a pinhole camera model and an effective focal length of $f = 800$ pixel.

5.1 Noise resilience

In our first experiment, we examine the influence of noise on the accuracy of the solution. To this end, we add Gaussian noise varying from 0 to 2 pixels to our measurements and execute 1000 runs for each level. Figure 5 indicates the



(a) (b) (c)

Fig. 5. Mean and median errors of roll (a), pitch (b) and yaw (c) angles for our 5-point, Stewénius' 5-point and our non-linear 6-point solutions.

resilience of our new 5-point algorithm and the one by Stewénius, both with model selection and 2-point rotation computation in case of small translations. We also display the results after 6-point non-linear refinement of our solutions. We can see that—due to the complexity of the computation—our 5-point algorithm has lower resilience to noise in comparison to Stewénius' solution. We can also see that the results after our non-linear refinement show highest mean absolute errors in all three rotational dimensions. The reason why the performance is degrading is related to bad rotation hypotheses under noise and hence convergence into wrong local minima. However, having a look at the median values of the error, we note that the result of our 6-point non-linear refinement still performs best. This is underlined by the fact that our 6-point non-linear refinement is able to identify a unique rotation, whereas the correct solution for the 5-point algorithms has been selected each time by hand. We also verified that our non-linear 6-point refinement remains robust in the special case of planar structure.

5.2 Behavior for small translations

We now proceed to analyzing the biggest advantage of our novel epipolar constraint. In order to investigate the influence of small translations on the accuracy of the algorithm, we run specific tests where the magnitude of the translation is varying from 0 to 3% of the average scene depth from viewpoint 1, namely 0.18. As mentioned in Section 3.3, we have to differentiate between cases of small translation and cases where the parallax is sufficiently high for running the 5-point algorithm. The same accounts for Stewénius' 5-point algorithm and notably every minimal solution that employs the essential matrix parametrization including the 8-point algorithm. Our proposed solution consists in taking the hypothesized rotations as starting points for our non-linear optimization. As shown in Figure 6, this yields very good results and even completely removes the error for small translations commited around the model selection boundary. The figure also displays the errors of pure 2-point and 5-point rotation computation without model selection. It proves that our novel epipolar constraint on the rotation only is indeed able to compute an exact rotation for small or even

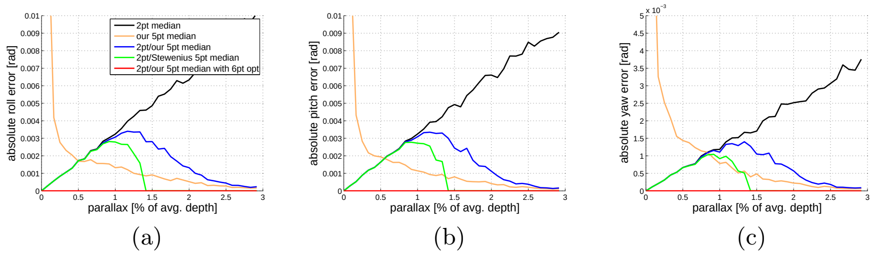


Fig. 6. Mean and median roll(a), pitch (b) and yaw (c) errors for small translation magnitudes.

zero translation magnitudes and does not suffer from numerical instability in this situation.

5.3 Performance in terms of accuracy and execution time

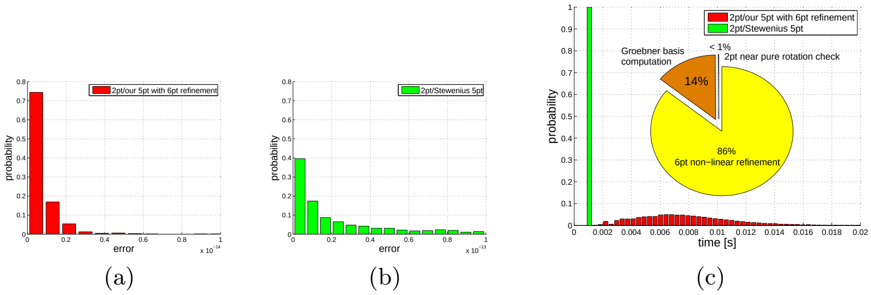
We also analyse the numerical accuracy and execution times of our new 6-point and Stewénius' 5-point methods. Figure 7 shows that—in the noise-free case—the numerical accuracy of our non-linear least squares solution is superior to Stewénius' least-squares solution by roughly one order of magnitude.

7(c) then finally shows the drawback of our method. The non-linear global minimization is pretty expensive and consumes 86% of the execution time. The overall execution time required by our method is currently still five times higher than the average time consumed by Stewénius' algorithm.

6 Conclusion

We have presented in this paper a new epipolar constraint for computing the rotation between two images independently of the translation. While the minimal 5-point computation of the rotation still suffers from the same model selection related inaccuracies as already existing approaches to epipolar geometry computation, the subsequent non-linear refinement of the hypotheses reduces the median error down to zero, even for vanishing translation magnitudes. The entire method thus represents a solution for computing a unique exact rotation and does not suffer from solution multiplicity or inaccuracies caused by the essential matrix parametrization. We also verified that our method does not suffer from planar degeneracies. The minimization is extendable to an arbitrary number of points, and can hence be used in any case where errors around the model selection boundary occur.

Once the rotation problem is solved, the translation between two viewpoints immediately follows as a simple two-point solution. Future work now consists in improving the efficiency of the algorithm and integrating it into a RANSAC scheme for continuous real-time frame-to-frame rotation estimation. The final



(a) (b) (c)

Fig. 7. Probability distribution over numerical accuracy for our proposed (a) and Stewénius' method (b). (c) shows the probability distribution over execution time for both methods. The pie chart shows the relative time consumption of the different parts of the algorithm.

goal is to check the impact of robust and translation decoupled rotation computation on structure-from-motion algorithms.

References

1. Nistér, D., Naroditsky, O., Bergen, J.: Visual odometry. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA (2004) 652–659
2. Davison, A., Reid, D., Molton, D., Stasse, O.: MonoSLAM: Real-time single camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **26**(6) (2007) 1052–1067
3. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR), Nara, Japan (2007)
4. Kneip, L., Chli, M., Siegwart, R.: Robust real-time visual odometry with a single camera and an IMU. In: Proceedings of the British Machine Vision Conference (BMVC), Dundee, Scotland (2011)
5. Fischler, M., Bolles, R.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6) (1981) 381–395
6. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **26**(6) (2004) 756–777
7. Stewénius, H., Engels, C., Nistér, D.: Recent developments on direct relative orientation. ISPRS Journal of Photogrammetry and Remote Sensing **60**(4) (2006) 284–294
8. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial Eigenvalue solutions to the 5-pt and 6-pt relative pose problems. In: Proceedings of the British Machine Vision Conference (BMVC), Leeds, UK (2008)
9. Longuet-Higgins, H.: Readings in computer vision: issues, problems, principles, and paradigms. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1987)

10. Kruppa, E.: Zur Ermittlung eines Objektes aus zwei Perspektiven mit innerer Orientierung. Sitzgsber. Akad. Wien, Math. Naturw. Abt., IIa. **122** (1913) 1939–1948
11. Torr, P., Fitzgibbon, A., Zisserman, A.: Maintaining multiple motion model hypotheses over many views to recover matching and structure. In: Proceedings of the International Conference on Computer Vision (ICCV), Bombay, India (1998) 485–491
12. Kalantari, M., Jung, F., Guedon, J.P., Paparoditis, N.: The five points pose problem: A new and accurate solution adapted to any geometric configuration. In: Proceedings of the Pacific Rim Symposium on Advances in Image and Video Technology, Tokyo, Japan (2009) 215–226
13. Lim, J., Barnes, N., Li, H.: Estimating relative camera motion from the antipodal-epipolar constraint. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **32**(10) (2010) 1907–1914
14. Cox, D.A., Little, J., O'Shea, D.: Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3/e (Undergraduate Texts in Mathematics). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2007)
15. Cayley, A.: About the algebraic structure of the orthogonal group and the other classical groups in a field of characteristic zero or a prime characteristic. Reine Angewandte Mathematik **32** (1846)
16. Thompson, E.: A method for the construction of orthogonal matrices. The Photogrammetric Record **3**(13) (1958) 55–59
17. Buchberger, B.: Multidimensional Systems Theory - Progress, Directions and Open Problems in Multidimensional Systems. Reidel Publishing Company, Dodrecht - Boston - Lancaster (1985)
18. Gebauer, R., Möller, H.M.: On an installation of Buchberger's algorithm. Journal of Symbolic Computation **6**(2-3) (1988) 275–286
19. Giovini, A., Mora, T., Niesi, G., Robbiano, L., Traverso, C.: "One sugar cube, please" or selection strategies in the Buchberger algorithm. In: Proceedings of the International Symposium on Symbolic and Algebraic Computation, Bonn, West Germany (1991) 49–54
20. Faugère, J.: A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra **139**(1–3) (1999) 61–88
21. Kukelova, Z., Bujnak, M., Pajdla, T.: Automatic generator of minimal problem solvers. In: Proceedings of the European Conference on Computer Vision (ECCV), Marseille, France (2008) 302–315
22. Arun, K., Huang, T., Blostein, S.: Least-Squares Fitting of Two 3-D Point Sets. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **9**(5) (1987) 698–700

Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) for the V-Charge project, and from the Swiss National Science Foundation under grant agreement n. 200021 125017/1.