

Absolute Pose for Cameras Under Flat Refractive Interfaces

Sebastian Haner and Kalle Åström

Centre for Mathematical Sciences, Lund University, Sweden

{haner,kalle}@maths.lth.se

Abstract

This paper studies the problem of determining the absolute pose of a perspective camera observing a scene through a known refractive plane, the flat boundary between transparent media with different refractive indices. Efficient minimal solvers are developed for the 2D, known orientation and known rotation axis cases, and near-minimal solvers for the general calibrated and unknown focal length cases. We show that ambiguities in the equations of Snell's law give rise to a large number of false solutions, increasing the complexity of the problem. Evaluation of the solvers on both synthetic and real data show excellent numerical performance, and the necessity of explicitly modelling refraction to obtain accurate pose estimates.

1. Introduction

Refractive structure-and-motion problems come in many varieties, depending on what relationships between scene structure, cameras and refractive planes are known. In many applications, such as in underwater photography where the camera views the world through a waterproof housing, the relationship between the camera and the glass can be pre-calibrated and assumed known. The back-projections of image points through and past the refractive interface can then be precomputed in the camera's coordinate system, and the whole assembly acts as an axial camera, as shown in [1]. Axial cameras are a special case of generalized cameras and algorithms previously developed for these can be used. For example, absolute pose for generalized cameras was solved minimally in [15] using three points, and relative pose in [16] using six points (as reported in [11], that algorithm degenerates for some axial camera configurations, but not in this case). To our knowledge, optimal two-view triangulation under refraction has not been solved, but standard linear methods are of course applicable when the back-projected rays are known. These three components along with refractive bundle adjustment [9] are the building blocks of a structure-and-motion system, and for the practically important case of known camera–refractive plane pose the prob-

lem can be considered more or less solved.

In [5] a theory is presented for the multiple-view geometry of cameras observing a scene through a common refractive interface. The existence of refractive projection, fundamental and homography matrices is shown and the relative pose problem is solved under very specific conditions. In [10] the relative translation problem is solved optimally under the L_∞ -norm, given the camera orientations.

In this paper we consider the problem of determining absolute pose of a camera observing known scene structure through a single known refractive planar interface. A more general problem was solved by [1] in the context of calibration, where the relative poses of camera, scene structure and refractive plane are all unknown, and indeed even the ratio of refractive indices. However, the algorithm requires at least eight point correspondences and has unnecessary degrees of freedom if the relative pose between scene structure and refractive plane is in fact known. In [4] the absolute pose problem is solved minimally with two point correspondences given that the camera's vertical direction is known, as given by an accelerometer. However, it is also assumed that the refractive plane is horizontal, significantly simplifying the problem. We present a minimal solution for the same case but with arbitrary refractive plane, and analyze the closed-form solutions to the known-orientation case. We also present a non-minimal algorithm for the general case using five points, and extend it to the case of unknown camera focal length using six points.

2. Snell's Law

Refraction of light at an optical medium boundary is described by Snell's law which states that

$$\rho_1 \sin \theta_1 = \rho_2 \sin \theta_2, \quad (1)$$

where $\rho_{1,2}$ are the refractive indices of the two media and $\theta_{1,2}$ the angles the impinging and refracted ray make with the surface normal (see Figure 1). Furthermore, the impinging ray with direction vector \vec{u} , the refracted ray \vec{v} and the plane normal \vec{n} must all lie in the same plane. Using properties of the cross product, Snell's law may then be expressed

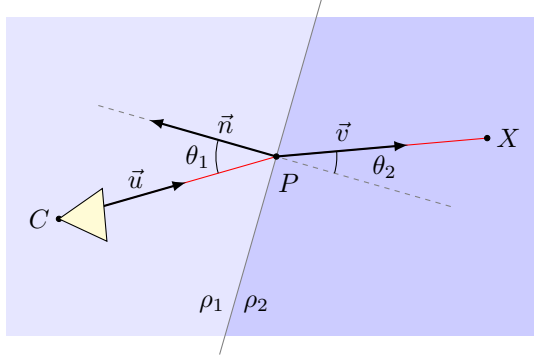


Figure 1. The image ray (C, \vec{u}) from the camera center intersects the plane with normal \vec{n} at P and is refracted into the ray (P, \vec{v}) according to Snell's law.

on vector form as

$$\rho_1 \frac{\vec{u} \times \vec{n}}{\|\vec{u}\| \|\vec{n}\|} = \rho_2 \frac{\vec{v} \times \vec{n}}{\|\vec{v}\| \|\vec{n}\|} \quad (2)$$

or equivalently

$$r \|\vec{v}\| (\vec{u} \times \vec{n}) = \|\vec{u}\| (\vec{v} \times \vec{n}), \quad (3)$$

where $r = \rho_1/\rho_2$. Note that by squaring both sides component-wise we obtain three equations which are polynomial in all variables, but since both sides of (3) are orthogonal to \vec{n} only two of them can be independent. The co-planarity constraint on the rays and normal can also be written as $\vec{u} \times \vec{v} \cdot \vec{n} = 0$, independently of the refractive indices. It is obvious that the camera center C and scene point X must also lie in this plane, implying

$$\vec{u} \times (X - C) \cdot \vec{n} = 0. \quad (4)$$

Given \vec{u} and \vec{n} , Snell's law also gives the refracted ray direction as

$$\vec{v} = r\vec{u} + (r \cos \theta_1 - \text{sign}(\cos \theta_1) \cos \theta_2) \vec{n} \quad (5)$$

where

$$\cos \theta_1 = -\vec{n} \cdot \vec{u} \quad (6)$$

and

$$\cos \theta_2 = \sqrt{1 - r^2(1 - \cos^2 \theta_1)}. \quad (7)$$

In what follows, we will usually assume that the intrinsic parameters of the camera are known so that the back-projected ray \vec{u} of an image point can be computed in the world coordinate system, given only the camera's orientation and translation. However, in the six-point algorithm presented in Section 8, the camera focal length is assumed unknown.

Furthermore, for the minimal solvers we only consider the case of a single flat refractive interface, since explicitly modeling two refractions using Snell's law leads to significantly more complex equations. For example, as shown in [1], computing the forward projection of a scene point into a camera through one refractive plane amounts to finding the roots of a fourth-degree polynomial, while doing the same for two parallel planes gives a 12th degree polynomial. The proposed non-minimal five- and six-point solvers only use the co-planarity constraints (4) and thus handle multiple refractions as long as all interfaces are parallel. In real applications there are usually two refractions, e.g. air-glass and glass-water, but if the glass is thin compared to the scene scale this is well-approximated by a single air-water refraction. This approximation is validated experimentally in Section 9.

3. Unknown Camera Translation

We start with the simpler problem of only finding the translation of the camera, with the orientation given. Snell's law (3) gives three equations per point, but since the projection only has two degrees of freedom, they are not independent and only give two constraints. To solve for the translation, one and a half points i.e. three coordinates, are thus needed; with two point matches, the extra constraint can be used to determine the refractive index ratio.

By coordinate transformation we may assume the refractive plane is described by $z = 0$, and that the image ray directions \vec{u} have been normalized to unit length and rotated into the global coordinate frame using the camera's known orientation. The intersection of the ray from the camera center C in the direction \vec{u} with the plane is then given by

$$P = C - \frac{C_z}{u_z} \vec{u} \quad (8)$$

and the refracted ray $\vec{v} = X - P$ where X is the corresponding known 3D point. Substituting this into (3) gives

$$r(\vec{u} \times \vec{n}) \|X - C + (C_z/u_z)\vec{u}\| = (X - C + (C_z/u_z)\vec{u}) \times \vec{n}. \quad (9)$$

Multiplying through by u_z , squaring both sides component-wise and using that $\vec{n} = (0, 0, 1)^T$ we obtain two polynomial equations (the z -component is identically zero) in the components of C and the squared ratio r^2 . The co-planarity constraint $(\vec{u} \times \vec{n}) \cdot (X - C) = 0$ provides two linear equations in the variables C_x and C_y which can be uniquely solved for as long as the two image rays \vec{u}_1 and \vec{u}_2 are not parallel. Substituting these values into Snell's law (9) gives two independent equations, linear in r^2 and quadratic in C_z . Eliminating r we obtain a quartic equation in C_z . The four solutions thus differ in the perpendicular distance C_z to the refractive plane and the refractive index ratio. If the ratio

r is known, this can be directly substituted into one of the equations quadratic in C_z , yielding two solutions. Given perfect data, the two equations have one common root corresponding to the true solution, but the other roots might not agree. This is possible because not all solutions returned are physically correct.

Note that Snell's law as stated in (1) only specifies the angle the refracted ray makes with the normal, but not on which side (see Figure 2), nor that the two rays should be on different sides of the plane. There is thus an ambiguity in the equations leading to incorrect solutions; some cases are illustrated in Figure 3. In the known refractive index case, experiments indicate one of the solutions is always physically incorrect, while in the unknown index case there can be between one and three valid solutions. False solutions can be filtered by checking if the back-projection ray given by (5) intersects the scene point.

This example illustrates a major difficulty in designing minimal solvers for refractive problems, namely an abundance of solutions which grows with the number of point matches required. When the orientation is not given the polynomial equations become much more involved and closed-form solutions are not possible.

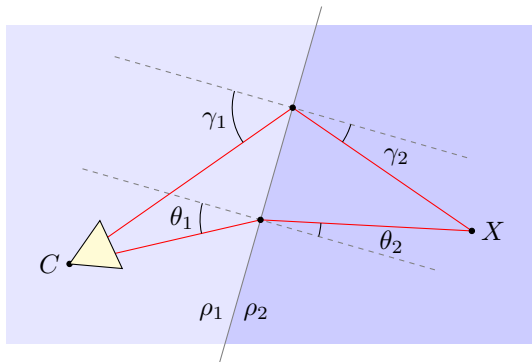


Figure 2. Ambiguity in Snell's law giving rise to false solutions. Both $\rho_1 \sin \theta_1 = \rho_2 \sin \theta_2$ and $\rho_1 \sin \gamma_1 = \rho_2 \sin \gamma_2$ are fulfilled.

4. Solving Polynomial Systems

A set of multivariate polynomials $f_i(\mathbf{x}) \in \mathbb{C}(\mathbf{x})$ generates an ideal I . If the polynomials have a finite number of common zeros, i.e. define a finite affine variety, the quotient space $\mathbb{C}(\mathbf{x})/I$ is isomorphic to \mathbb{C}^k , where k is the number of solutions to the system. We can then select a k -dimensional basis \mathcal{B} for $\mathbb{C}(\mathbf{x})/I$ and express the linear map $T_p: f(\mathbf{x}) \mapsto p(\mathbf{x})f(\mathbf{x})$ as a k -by- k matrix M_p , known as the action matrix for the polynomial $p(\mathbf{x})$. If we can find this matrix for any polynomial p , usually chosen as just one of the variables, its eigenvectors give the basis monomials in \mathcal{B} evaluated at the solutions. A basis for the quotient space can be computed using algebraic geometry tools such as Macaulay2 [8] or Maple which then also compute the

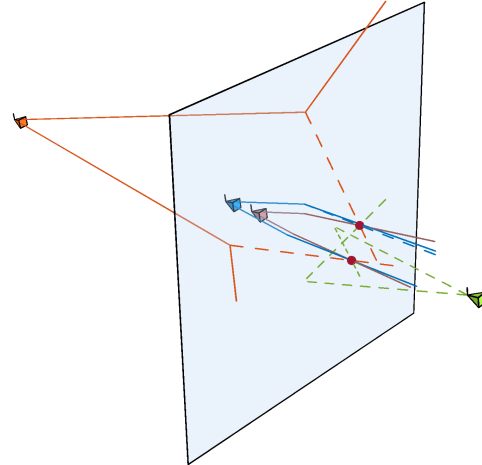


Figure 3. Four solutions to the known orientation, unknown refractive index case, three of which are incorrect due to ambiguities in the equations. Solid lines are the physical back-projections of the image points while dashed lines illustrate spurious optical paths consistent with (3) giving rise to false solutions.

number of solutions. These systems rely on rational arithmetic to compute a Gröbner basis for the ideal; if the coefficients of the polynomials are given as inexact floating point numbers, cancellations might not be detected in the elimination and a larger basis than necessary may be returned. In some cases one can generate “ground truth” problem instances with rational coefficients to avoid this, but this is very difficult with Snell's law. Therefore we will only be able to give an upper bound on the number of solutions to the polynomial systems that follow, which may or may not be reached depending on the input data.

In practice, to compute the action matrix we first generate an expanded set of equations by multiplying the f_i by monomials. The expanded system may be written as $\mathbf{A}\mathbf{X} = \mathbf{0}$ where \mathbf{A} is a coefficient matrix and \mathbf{X} a vector of monomials, partitioned into excessive, reducible and basis monomials. The goal is to express the reducible monomials, which consist of the basis monomials multiplied by the action variable, in terms of the basis monomials, giving us the action matrix. This is achieved by eliminating the excessive monomials using Gaussian elimination or QR factorization. If the equation set was sufficiently expanded, it should now be possible to solve for the action matrix linearly. However, if the system has fewer solutions than assumed, correlations between coefficients may lead to a rank deficiency at this stage. This happens in our five- and six-point solvers and we deal with this as explained in Section 7.

A different approach is to formulate the equations as a *polynomial eigenvalue problem*. Choosing a “hidden” variable z , any polynomial system can be written as

$$(z^n A_n + z^{n-1} A_{n-1} + \dots + z A_1 + A_0)\mathbf{X} = \mathbf{0}, \quad (10)$$

where \mathbf{X} are monomials not containing z . If the coefficient matrices are square and A_n or A_0 have full rank, the system can be solved as a generalized eigenvalue problem. For more background on algebraic geometry, the action matrix method and polynomial eigenvalue problems, see [6], [3] and [14].

5. The 2D Case

In two dimensions there is only one rotation angle for the camera orientation which greatly simplifies the equations. The coordinate system may be transformed so that the known refraction interface, now just a line, coincides with the y -axis. The intersection P of an image ray \vec{u} with the line is then given by

$$P = C - \frac{C_y}{R_{:,2} \cdot \vec{u}} R^\top \vec{u}, \quad (11)$$

where

$$R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (12)$$

and $R_{:,2}$ denotes the second column of the camera's rotation matrix. Embedding in 3D and plugging into Snell's law (3) now only provides one constraint per projection, so three points are needed. Multiplying with the denominator in P and squaring as before, we obtain three polynomials of total degree six in the variables $C_x, C_y, c = \cos \theta$ and $s = \sin \theta$, and add the constraint $c^2 + s^2 = 1$.

Analysis of the resulting system using algebraic geometry tools shows it may have up to 96 solutions. However, there is symmetry in the equations; s and c only occur in even powers meaning that if (C_x, C_y, c, s) is a solution, so is $(C_x, C_y, -c, -s)$. This corresponds to the fact that rotating the camera 180° results in the same image projections in 2D. This symmetry can be exploited when solving the system using the action matrix method [2], essentially allowing us to solve for only half of the solutions giving an effective basis size of 48. We multiply the four original equations with all monomials which are products of the 'basis monomials' c^2, s^2, cs, C_x, C_y s.t. the total degree does not exceed 7 and the degrees of the variables c, s, C_x, C_y do not exceed 6, 5, 7 and 3 respectively (of course this assignment is symmetric in c and s and C_x and C_y). This results in 1272 equations in 1484 monomials. Using c^2 as the action monomial then allows us to solve for C_x, C_y, c^2, s^2 and cs . We also know that $c > 0$, since otherwise the optical axis is pointing away from the refractive plane, so the sign of s can be determined from cs .

Experiments indicate that there is rarely more than one or two physically correct solutions among the 48. An optimized Matlab implementation of the solver runs in 80 ms including Newton steps to refine the solutions. However,

since the 2D case is of limited practical importance we will not consider this solver further in this paper.

6. Known Rotation Axis

The problem in 3D of known translation and one degree of rotational freedom is similar to the pure 2D pose problem, which is a special case. This problem formulation arises when e.g. the camera's elevation and roll angle can be determined using an accelerometer, such as in a mobile phone, but the azimuth is unknown. A special case was considered in [4] where it was assumed that the refractive plane is horizontal, i.e. that the normal is known.

We transform the coordinate system so that the unknown rotation axis is parallel with the y -axis. If the refractive plane is described by $\vec{n} \cdot \mathbf{X} + d = 0$ the intersection with an image ray is given by

$$P = C - \frac{\vec{n} \cdot C + d}{\vec{n} \cdot (R^\top \vec{u})} R^\top \vec{u}, \quad (13)$$

where

$$R = R_{xz} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \quad (14)$$

is the camera rotation matrix, decomposed into the known elevation and roll and the unknown y -axis rotation. Four degrees of freedom means two projections are required to solve the minimal case, and as usual plugging into Snell's law gives four independent polynomial equations after multiplying with the denominator in P and squaring. Along with the constraint $\cos^2 \theta + \sin^2 \theta = 1$, these are enough to solve the problem using the action matrix technique, and analysis with Macaulay2 shows there could be up to 64 solutions. However, there is no longer symmetry in the rotation parameters that can be used to reduce the basis size, and it turns out the equation set has to be expanded to thousands of polynomials, yielding a slow solver (~ 1 s). Instead we change the rotation parametrization to that used in [13, 4], letting

$$q = \tan(\theta/2) \quad (15)$$

giving

$$\cos \theta = (1 - q^2)/(1 + q^2) \quad (16)$$

$$\sin \theta = 2q/(1 + q^2). \quad (17)$$

The resulting system can now be solved as a polynomial eigenvalue problem (PEP). To transform it to PEP form the equation set has to be expanded, but we use a simpler strategy than the resultant-based method proposed in [14]. We expand the original system, consisting of all six equations from (3) and both co-planarity constraints (4), to 32

equations by multiplying with monomials C_x , C_y and C_z . It may seem unnecessary to include the co-planarity constraints since they are implicit in Snell's law, but in transforming (3) to polynomial form, some information is lost which is retained in (4), further constraining the solutions. Hiding the variable q we obtain a matrix polynomial equation

$$(q^8 A_8 + q^7 A_7 + \dots + q A_1 + A_0) \mathbf{X} = \mathbf{0} \quad (18)$$

of degree eight where the A_i are 32-by-20 matrices and \mathbf{X} a vector of 20 monomials in $C_{x,y,z}$. To convert this to a PEP the A_i must be made square without losing rank, and this is accomplished by left-multiplying (18) by a random 20-by-32 matrix or simply by A_1^\top , similar to what was done in [7].

An upper bound on the number of solutions to the original system is 112, while the PEP returns up to $8 \cdot 20 = 160$ solutions. Only a subset fulfill the equations and only a handful provide physically correct solutions.

6.1. Degeneracies

As noted in Section 5, the 2D case requires a minimum of three points. This means that if the scene points lie in the plane spanned by the camera up vector and the refractive plane normal, the translation cannot be solved for using only two points. Note that this is a degeneracy shared with the non-refractive case.

The solver itself also exhibits degeneracies even when the problem is well-posed; if the 3D points $X_{1,2}$, the plane normal \vec{n} and the camera center lie in the same plane, or if $X_1 - X_2$ is parallel with \vec{n} , the solver will fail. These conditions are however unlikely to be exactly fulfilled by real data. In addition, the parametrization chosen has a singularity at $\theta = 180^\circ$, which in most situations can be avoided by suitable rotation of the coordinate system around the y -axis.

7. Absolute Pose with Five Points

Three points are minimal for the general absolute pose problem, but using the same approach as above the equations become too difficult, with thousands of terms. As was noted in [1] much information can be gained using only the co-planarity constraints (4), given enough point correspondences. In that paper, absolute pose and refractive plane parameters are solved for linearly using 11 points, and with eight points using a clever application of a minimal solver for the standard five-point relative pose problem. This is afforded by the relative simplicity of the co-planarity constraints compared to the full Snell's law. We therefore solve the absolute pose problem using only the co-planarity constraints, which requires five point correspondences. From these equations all parameters except for the perpendicular distance of the camera to the plane can be recovered.

To simplify the equations, we may assume that the refractive plane normal is parallel with the z -axis, and the co-planarity constraints then take the form

$$R^\top \vec{u} \times (X - C) \cdot (0, 0, 1) = 0. \quad (19)$$

Note that this equation does not contain C_z . We parametrize the camera rotation matrix R using quaternions $\mathbf{q} = (s, \vec{\omega})$ so that

$$R(\mathbf{q}) = 2(\vec{\omega} \vec{\omega}^\top - s[\vec{\omega}]_\times) + (s^2 - \vec{\omega}^\top \vec{\omega}) I \quad (20)$$

where $[\cdot]_\times$ is the cross product matrix s.t. $[\vec{a}]_\times \vec{b} = \vec{a} \times \vec{b}$. $R(\mathbf{q})$ is only orthonormal if \mathbf{q} has unit length, but all matrix elements scale with $\|\mathbf{q}\|^2$. Since (19) is homogeneous in R , the unit-length requirement can be dropped and we set the scalar component $s = 1$, as was done in [16]. Since both \mathbf{q} and $-\mathbf{q}$ represent the same rotation, fixing the sign and magnitude in this way gives a minimal parametrization and halves the number of solutions, at the cost of introducing a singularity for all 180° rotations (for which $s = 0$). We now have five polynomial equations of total degree three in five unknowns, and analysis gives an upper bound on the number of solutions as 48. By multiplying with all monomials up to total degree three, 280 equations are obtained and the action matrix method can be applied. As it turns out, correlations in the coefficients of the system means there are in general only 16 solutions. This manifests itself in the action matrix algorithm as a rank deficiency of four when solving for the reduction monomials in terms of the basis monomials. This means there are four monomials which cannot be solved for and must be removed from the basis. However, which monomials to remove depends on the data, and can be determined using column-pivoting QR factorization, which heuristically partitions the columns of the coefficient matrix into a well-conditioned set and four which are linearly dependent on the others.

The above solution works independently of the refractive indices or indeed how many refractive layers are traversed, as long as each ray stays in a single plane. Assuming there is only one interface given by $z = 0$ and that the refractive index ratio is known, we can plug the rotation and x - and y -translation into the full Snell's law for a single projection and obtain a quadratic equation for the z -translation. The two roots correspond to the situation in Figure 2 and it can be shown that the physical solution is the one with the camera closest to the plane. If there are several parallel refractive planes and/or the refractive indices are unknown, the method presented in [1] can be used to solve for the translation, given enough point correspondences.

7.1. Degeneracies

If all points lie on a line parallel with the plane normal, the camera can be rotated around this line without chang-

ing the image projections (in fact there is even a three-dimensional family of solutions since the constraints are weaker). The solver will also fail if all points and the plane normal lie in a common plane. The singularity of the parametrization can be avoided with high probability by randomly rotating the coordinate system about the z -axis.

8. Unknown Focal Length with Six Points

The five-point formulation above is easily extended to the case of unknown camera focal length. Under the same assumptions the co-planarity constraints take the form

$$R^T K^{-1} \vec{u} \times (X - C) \cdot (0, 0, 1) = 0, \quad (21)$$

where

$$K^{-1} = \text{diag}(1, 1, f) \quad (22)$$

and f is the focal length. The extra degree of freedom means six points are required, and analysis of the system gives an upper bound on the number of solutions as 104. Note however that a symmetry has been introduced; changing the sign of the focal length and rotating the camera 180° around the optical axis results in the same image projections. With the chosen parametrization, such a rotation is equivalent to flipping the signs of the first two vector components of the quaternion, and there is thus a two-fold symmetry in the variables f , ω_1 and ω_2 .

We obtain 648 equations by multiplying the original six with all monomials which are products of the ‘basis monomials’ f^2 , ω_1^2 , ω_2^2 , $f\omega_1$, $f\omega_2$, $\omega_1\omega_2$, ω_3 , C_x , C_y s.t. the degrees of the variables f , ω_1 , ω_2 , ω_3 , C_x , C_y do not exceed 3, 3, 3, 1, 1 and 1 respectively. Choosing ω_3 as the action variable, a basis size of 52 is now sufficient to compute the solution using the same basis selection method as for the five-point solver. Since the monomials f^2 , $f\omega_1$ and $f\omega_2$ are in the basis, the correct signs for the rotation components are easily deduced.

8.1. Degeneracies

While the five-point solver fails if the plane normal and scene points lie in a common plane, the equations (21) of the six-point problem are actually under-determined in this case. The problem as such is still well-posed, but the coplanarity conditions are not enough to constrain the solution.

9. Experiments

The solvers were implemented in Matlab, and all experiments run on a 3.0 GHz Core 2 Duo computer. The known-axis solver runs in around 60 ms, most of which is spent solving the rather large PEP problem using Matlab’s `polyeig` command. In the action matrix-based solvers

most of the time is spent in the elimination step reducing the expanded system, and when using sparse fill-reducing QR factorization the five- and six-point solvers run in around 10 and 20 ms respectively. We test the solvers’ numerical stability using randomly generated problem instances. As seen in Figure 4, the known-axis and five-point solvers perform very well with essentially no failure cases, while the six-point solver is somewhat more unstable. To see if the degeneracies inherent to the solvers is problematic, we also generate random degenerate configurations of plane and scene points, and disturb the points slightly by adding normally distributed noise of relative magnitude 10^{-5} . Figure 5 shows that the solvers still manage to find solutions in the majority of cases, indicating that the set of problematic problem instances is small and not of practical concern.

Figure 6 shows the distribution of the number of real solutions returned by the solvers. Among these, the five- and six-point solvers never returned more than one physically correct solution, and the known-axis solver never more than three.

The known-axis solver was derived under the assumption that there is only one refractive interface. In situations where the two media are separated by e.g. a sheet of glass, this assumption introduces an error. To quantify this we conduct a synthetic experiment where a sheet of glass is placed roughly half-way between a camera in air and scene points in water, which are around ten length units apart. The refractive index of air is taken to be unity, water 1.333 and glass 1.5. Figure 7 shows the translational and angular error of the pose estimate for varying glass thicknesses and levels of image measurement noise. It is clear that the error introduced by the approximation is small and is dominated by the image measurement error.

9.1. Real Data

To validate our algorithms on real data, a Rubik’s cube was submerged in a small rectangular acrylic plastic tank with clear sides. The cube was captured by an HTC Desire mobile phone while recording the accelerometer readings. The relative locations of the cube corners and the refractive plane (the tank wall) was measured by ruler, and the image correspondences marked by hand. While the five- and six-point solvers only return one physically valid solution given perfect synthetic data, several plausible solutions with small reprojection errors may be found with noisy input. To determine the best camera pose from each solver a RANSAC-like approach was used, where minimal sets of corner matches were selected at random, and all valid solutions compared in terms of reprojection error, computed over all points. While the solvers neglect the effect of the tank wall, this is included when computing the reprojection errors. The index of refraction of water was taken as 1.333, and 1.49 for the plastic. In addition to the three proposed

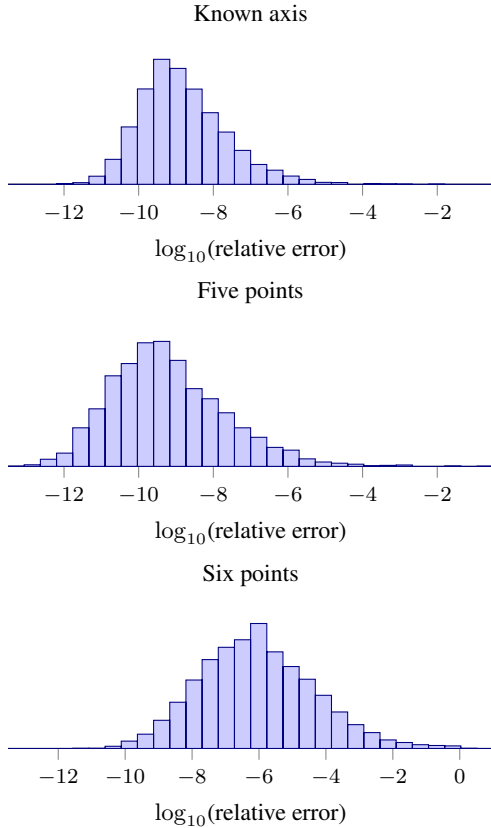


Figure 4. Distribution of solver error relative to ground truth, computed over 5000 random problem instances.

solvers, we also solve for the pose while ignoring the refraction effects, i.e. assuming all refractive indices are unity.

Figure 8 shows the reprojections of the different algorithms overlaid on two of the images, and their reprojection errors are summarized in Table 1. The average error in the

Solver	2-pt	5-pt	6-pt	No refraction	Iterative
Error	16.5	7.1	8.3	31.9	5.3

Table 1. Average reprojection error magnitude in pixels over several runs with different random seeds to the RANSAC procedure. Images were captured at 2592×1952 resolution. The iterative solution minimizes the reprojection error over all points seeded with the five-point solution, and represents a lower bound on the error.

focal length returned by the six-point solver was 67 pixels or 2.6% compared with the ground truth camera calibration. The reconstructed camera poses are shown in Figure 9. The five- and six-point solutions agree closely while the two-point solution shows translation errors in the vertical direction, probably due to noisy or biased accelerometer data. The no-refraction assumption clearly leads to large reprojection errors and skewed pose estimates.

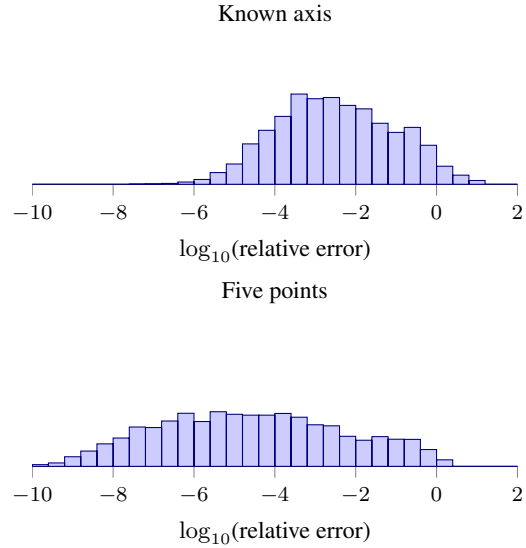


Figure 5. Distribution of solver error computed over 5000 nearly degenerate random problem instances.

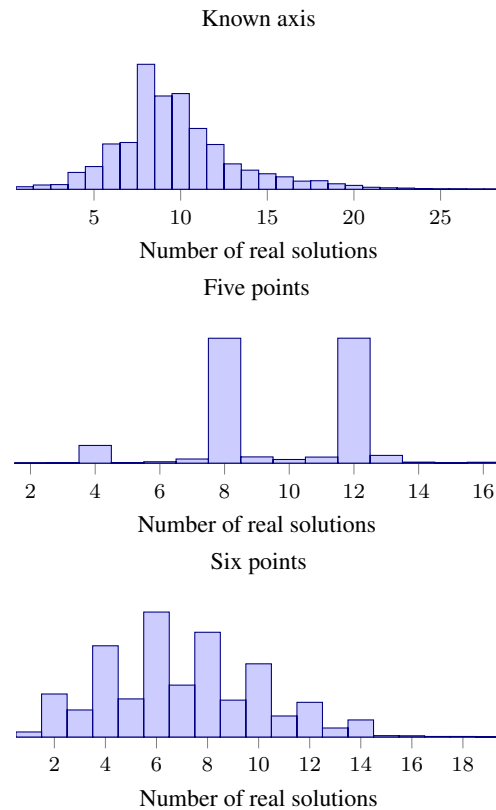


Figure 6. Distribution of the number of real solutions returned by the solvers, computed over 5000 random problem instances.

10. Conclusions

We have presented efficient solutions to several variants of the refractive absolute pose problem. We have shown

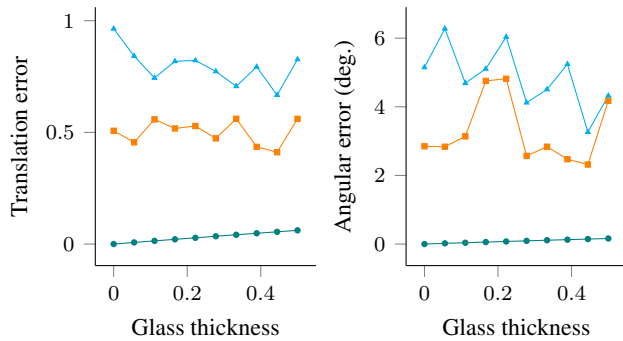


Figure 7. Pose error of the known rotation axis solver as a function of glass thickness and image noise, averaged over 100 random problem instances (scene scale approx. 10 units). Bottom, middle and top graphs correspond to zero, one and two pixel std. dev. Gaussian noise respectively.

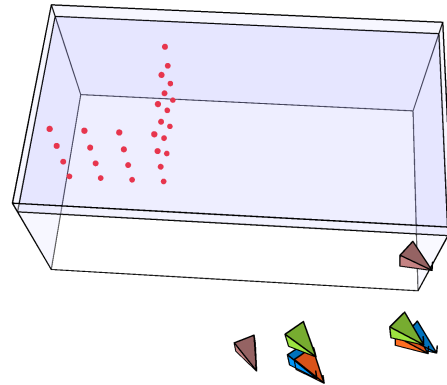


Figure 9. Reconstructed poses from two images of the Rubik's cube experiment. Green: known-axis, orange: 5-point, blue: 6-point, mauve: no-refraction solution.

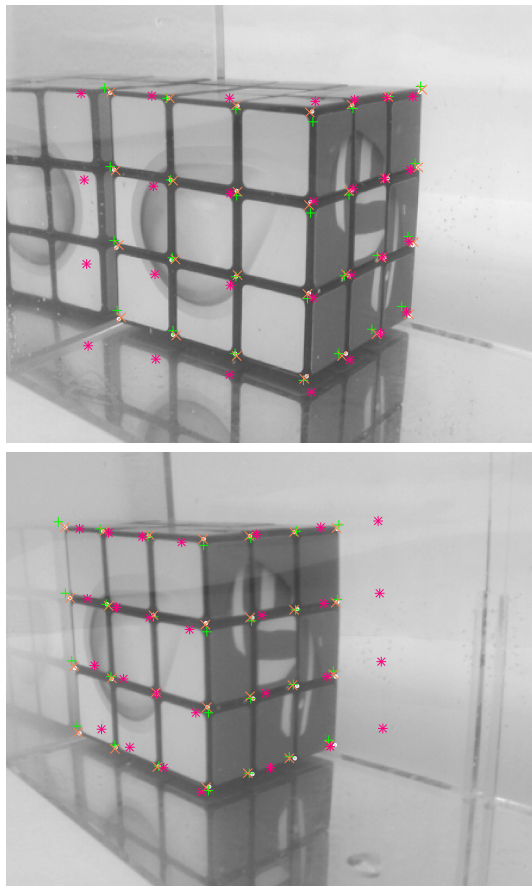


Figure 8. Visualization of the reprojection errors in the Rubik's cube experiment. Manual image measurements of the cube corners are shown as white dots. The known-axis solver reprojection is shown as green plus-signs, the five-point solver as orange crosses, and the reference non-refractive solution as magenta stars. The reprojections of the six-point solver are very similar to the five-point solution and are omitted for clarity.

that the solvers are numerically stable and produce accurate results on real images. There is still room for improvement with regard to numerical stability, particularly for the six-point solver, and with regard to speed. For example, techniques from [12] could be used to reduce the size of the expanded equation sets used in the action matrix method. Degeneracies for the problem setups have also not been thoroughly explored. While the goal of a truly minimal solver in the general case has not yet been reached, the presented algorithms are likely to be faster, enough to compensate for the higher number of iterations required in a hypothesize-and-test framework. The unmanageable size of the polynomials derived from Snell's law in the general case suggests a new approach is needed, where the physical constraints can be enforced to constrain the number of solutions.

All the solvers presented in the paper are available as Matlab implementations at http://github.com/sebhaner/refractive_pose.

References

- [1] A. Agrawal, S. Ramalingam, Y. Taguchi, and V. Chari. A theory of multi-layer flat refractive geometry. In *Conference on Computer Vision and Pattern Recognition*, pages 3346–3353. IEEE, 2012. 1, 2, 5
- [2] E. Ask, Y. Kuang, and K. Åström. Exploiting p-fold symmetries for faster polynomial equation solving. In *International Conference on Pattern Recognition*, pages 3232–3235. IEEE, 2012. 4
- [3] M. Byröd, K. Josephson, and K. Åström. Fast and stable polynomial equation solving and its application to computer vision. *International Journal of Computer Vision*, 84(3):237–256, 2009. 4
- [4] Y. Chang and T. Chen. Multi-view 3d reconstruction for scenes under the refractive plane with known vertical direction. In *International Conference on Computer Vision*, 2011. 1, 4

- [5] V. Chari and P. F. Sturm. Multi-view geometry of the refractive plane. In *British Machine Vision Conference*, 2009. 1
- [6] D. A. Cox, J. Little, and D. O’Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3rd ed.* Springer-Verlag New York, Inc., 2007. 4
- [7] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Conference on Computer Vision and Pattern Recognition*, pages 125–132. IEEE, 2001. 5
- [8] D. R. Grayson and M. E. Stillman. Macaulay2, a software system for research in algebraic geometry. Available at <http://www.math.uiuc.edu/Macaulay2/>. 3
- [9] A. Jordt-Sedlazeck and R. Koch. Refractive structure-from-motion on underwater images. In *International Conference on Computer Vision*, pages 57–64, Dec 2013. 1
- [10] L. Kang, L. Wu, and Y.-H. Yang. Two-view underwater structure and motion for cameras under flat refractive interfaces. In A. W. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, editors, *European Conference on Computer Vision*, volume 7574 of *Lecture Notes in Computer Science*, pages 303–316. Springer, 2012. 1
- [11] J.-H. Kim, H. Li, and R. I. Hartley. Motion estimation for nonoverlapping multicamera rigs: Linear algebraic and L_∞ geometric solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1044–1059, 2010. 1
- [12] Y. Kuang and K. Åström. Numerically stable optimization of polynomial solvers for minimal problems. In A. W. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, editors, *European Conference on Computer Vision*, volume 7574 of *Lecture Notes in Computer Science*, pages 100–113. Springer, 2012. 8
- [13] Z. Kukelova, M. Bujnak, and T. Pajdla. Closed-form solutions to minimal absolute pose problems with known vertical direction. In R. Kimmel, R. Klette, and A. Sugimoto, editors, *Asian Conference on Computer Vision*, volume 6493 of *Lecture Notes in Computer Science*, pages 216–229. Springer, 2010. 4
- [14] Z. Kukelova, M. Bujnak, and T. Pajdla. Polynomial eigenvalue solutions to minimal problems in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1381–1393, 2012. 4
- [15] D. Nistér. A minimal solution to the generalised 3-point pose problem. In *Conference on Computer Vision and Pattern Recognition*, pages 560–567, 2004. 1
- [16] H. Stewénius, D. Nistér, M. Oskarsson, and K. Åström. Solutions to minimal generalized relative pose problems. In *Workshop on Omnidirectional Vision*, 2005. 1, 5