

Optimal Estimation of Perspective Camera Pose

Carl Olsson

calle@maths.lth.se

Fredrik Kahl

fredrik@maths.lth.se

Magnus Oskarsson

magnuso@maths.lth.se

Centre for Mathematical Sciences,
Lund University, Sweden

Abstract

In this paper we propose a practical and efficient method for finding the globally optimal solution to the problem of camera pose estimation for calibrated cameras. While traditional methods may get trapped in local minima, due to the non-convexity of the problem, we have developed an approach that guarantees global optimality.

*The scheme is based on ideas from global optimization theory, in particular, convex under-estimators in combination with branch and bound. We provide a provably optimal algorithm and demonstrate good performance on both synthetic and real data.*¹

1 Introduction and Problem Formulation

One of the basic problems both in computer vision and in photogrammetry is the pose estimation problem. Given a number of correspondences between points in 3D space and the images of these points, the pose estimation problem consists of estimating the rotation and position of the camera. Here the camera is assumed to be calibrated. A typical application is shown in Figure 1, where the object is to relate a camera to a scanned 3D model of a chair.

This problem has been studied for a long time. The minimal amount of data required to solve the problem is three points and in this case there are up to four solutions. This result has been shown a number of times, but the earliest solution is to our knowledge due to Grunert, already in 1841, [2]. A good overview of the minimal solvers and their numerical stability can be found in [3]. Given at least six point correspondences the predominant method to solve the pose estimation problem is by using the DLT algorithm as a linear starting solution for a gradient descent method, see [4]. A global method for uncalibrated camera pose can be found in [5] and for the registration problem, see [7]. There are also quasi-linear methods that give a unique solution given at least four point correspondences, e.g. [8].

¹This work has been funded by the European Commission's Sixth Framework Programme (SMERobot grant no. 011838), the VISCOS project funded by the Swedish Foundation for Strategic Research, and by the Swedish Research Council (grants no. 2004-4579, and no. 2005-3230).



Figure 1. Chair experiment with 3D-scanner

These methods solve a number of algebraic equations, and do not minimize the reprojection error.

In this paper we develop an algorithm that minimizes the norm of the reprojection errors, that is, a non-linear least-squares problem. Given a Gaussian noise model for the image measurements and assuming i.i.d., the ML estimate is obtained. Throughout the paper, a calibrated pinhole camera model is utilized, see e.g. [4]. Given m world points X_i (represented by 3-vectors) and corresponding image points $x_i = [x_i^1 \ x_i^2]^T$, we want to find the camera translation t and rotation R that minimizes f :

$$f(R, t) = \sum_{i=1}^m d(x_i, \pi(X_i))^2 = \sum_{i=1}^m \left(\left(x_i^1 - \frac{r_1^T X_i + t_1}{r_3^T X_i + t_3} \right)^2 + \left(x_i^2 - \frac{r_2^T X_i + t_2}{r_3^T X_i + t_3} \right)^2 \right), \quad (1)$$

where $\pi(\cdot)$ is the perspective projection with $R^T = [r_1 \ r_2 \ r_3]$ and $t^T = [t_1 \ t_2 \ t_3]$.

We will in the next section show how to derive a branch and bound algorithm that finds the global optimum for (1).

2 A Branch and Bound Algorithm for the Camera Pose Estimation Problem

In this section we derive a simple branch and bound algorithm (cf. [6]) for the camera pose problem using the technique of convex under-estimators

Branch and bound algorithms are iterative methods for finding global optima of non-convex problems. They work by calculating sequences of provably lower bounds, which converge to the global minimum. The result of such an algorithm is usually an ϵ -suboptimal solution, i.e., a solution

that is at most ϵ from the global minimum for a predetermined value of ϵ .

Consider the following problem. We want to minimize a non-convex function $f(x)$ over a rectangle D_0 . For any sub-rectangle $D_n \subseteq D_0$, let $f_{min}(D_n)$ be the minimum value of f on D_n and let $f_{low}(D_n)$ be a lower bound for f on D_n . We require that the approximation gap $f_{min}(D_n) - f_{low}(D_n)$ goes uniformly to zero as the maximum length of the sides of D_n goes to zero. If such a lower-bounding function can be obtained then a strategy to obtain an ϵ -suboptimal solution is to divide the domain into rectangles, such that the approximation gap is less than ϵ everywhere, and compute f_{low} in each rectangle. However the number of such rectangles increases rapidly and therefore this may not be feasible. To avoid this problem a strategy to create as few rectangles as possible can be deployed. Assume that we know that $f_{min}(D_0) < k$. If $f_{low}(D_n) > k$ for some n then there is no point in refining D_n further since the minimum will not be attained in D_n . Thus D_n and all $D_k \subseteq D_n$ can be discarded.

Now we will derive a parametrization of the objective function for the camera pose estimation problem. Recall that the problem is to find a rotation R and a translation t such that (1) is fulfilled. A common way to parametrize rotations is to use quaternions (see [1]). Let $q = (q_1, q_2, q_3, q_4)^T$ be the unit quaternion parameters of the rotation matrix R . We note that when parametrizing with quaternions, equation (1) can be rewritten as

$$f(q, t) = \sum_{i=1}^{2m} \left(\frac{q^T A_i q + a_i^T t}{q^T B_i q + b_i^T t} \right)^2, \quad (2)$$

where A_i and B_i are 4×4 -matrixes, a_i and b_i 4×1 -vectors, determined by the data points x_i and X_i . This is a non-convex rational function in seven variables.

To obtain lower bounds on this function we proceed by formulating a convex optimization problem for which the solution gives a lower bound. The quadratic forms $q^T A_i q + a_i^T t$ and $q^T B_i q + b_i^T t$ contain terms of the form $q_i q_j$. Therefore we introduce the new variables $s_{ij} = q_i q_j$ or equivalently:

$$s_{ij} \leq q_i q_j, \quad (3)$$

$$s_{ij} \geq q_i q_j, \quad (4)$$

$i = 1, \dots, 4, j = 1, \dots, 4$. The constraints (4) are convex if $i = j$ and (3) is not convex for any i, j . If we replace $q_i q_j$ in (3) with its concave envelope (see [9]) and $q_i q_j$ in (4) by its convex envelope then (3) and (4) will be convex conditions. By doing this we expand the domain for s_{ij} and thus the minimum for this problem will be lower or equal to the original problem. The relaxed versions of the equations

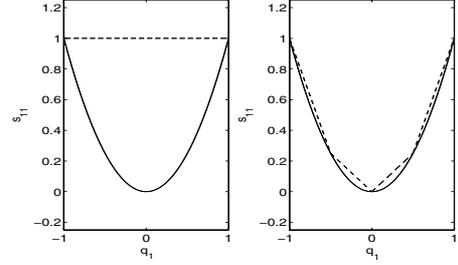


Figure 2. Upper and lower bounds of s_{11} , which relaxes q_1^2 in the interval $[-1, 1]$. **Left: the initial bound. Right: when the interval has been divided four times. Note the lower bound is exact since $q_1^2 \leq s_{11}$ is convex.**

(3) and (4) for $i \neq j$ is

$$-s_{ij} + q_i q_j^U + q_i^U q_j - q_i^U q_j^U \geq 0, \quad (5)$$

$$-s_{ij} + q_i q_j^L + q_i^L q_j - q_i^L q_j^L \geq 0, \quad (6)$$

$$s_{ij} - (q_i q_j^L + q_i^U q_j - q_i^U q_j^L) \geq 0, \quad (7)$$

$$s_{ij} - (q_i q_j^U + q_i^L q_j - q_i^L q_j^U) \geq 0. \quad (8)$$

If $i = j$ then the concave envelope of (3) is simply a line $a q_i + b$, where a and b are determined by noting that the values $(q_i^L)^2$ and $(q_i^U)^2$ should be attained at the points $q_i = q_i^L$ and $q_i = q_i^U$, respectively. Figure 2 shows the upper and lower bounds of s_{11} when $-1 \leq q_1 \leq 1$. We see that even when the interval has only been divided four times the upper bound is quite close to the lower bound. This gives some indication on the convergence speed of the lower bounds.

The terms $(q^T A_i q + a_i^T t)^2$ and $(q^T B_i q + b_i^T t)^2$ in the objective function (2) can now be rewritten as $(\hat{A}_i \hat{q})^2$ and $(\hat{B}_i \hat{q})^2$ respectively. Here \hat{q} is a 13×1 vector containing the parameters $(t_1, \dots, t_3, s_{11}, s_{12}, \dots, s_{44})$ and \hat{A}, \hat{B} are 13×13 matrices. The expression $(\hat{A}_i \hat{q})^2$ is the square of a linear function and is therefore convex.

It is well known that a rational function with linear denominator and quadratic numerator of the form x^2/y for $y > 0$ is convex. Therefore by replacing the denominator with the affine function $c_i(\hat{B}_i \hat{q}) + d_i$ (that is, the concave envelope of $(\hat{B}_i \hat{q})^2$), a convex under-estimator of the original cost-function is obtained. The constants c_i and d_i are determined from the bounds on q_i and t_i . The full convex function is then

$$f_{low}(\hat{q}) = \sum_{i=1}^{2m} \frac{(\hat{A}_i \hat{q})^2}{c_i(\hat{B}_i \hat{q}) + d_i}. \quad (9)$$

Minimizing this function subject to the constraints on s_{ij} yields a lower bound on the original objective function (1). At the same time, one can compute the actual value of the objective function f . If the difference is small (less than ϵ), one can stop. As the intervals on $q_i, i = 1, \dots, 4$ are divided

into smaller ones, it can be shown that the lower bound f_{low} converges uniformly to the function (2), cf. [9].

To simplify the problem further, we make the following modifications. Without loss generality, one can choose the world coordinate system such that $X_i = [0 \ 0 \ 0]^T$ for some i , since the cost-function is independent of the world coordinate frame. Further, as the cost-function is a rational function of homogeneous quantities, it can be dehomogenized by setting $t_3 = 1$. Thus, for point i we obtain

$$d(x_i, \pi(X_i))^2 = (x_i^1 - t_1)^2 + (x_i^2 - t_2)^2. \quad (10)$$

Note that this is a convex function. Further, we can restrict the search space by enforcing a maximum error bound on the reprojection error for this point, say γ_{max} pixels. This results in bounds on t_j such that $x_i^j - \gamma_{max} < t_j < x_i^j + \gamma_{max}$ for $j = 1, 2$.

Since t_3 can be geometrically interpreted as the distance from the camera centre to the point X_i (along the optical axis), we have effectively normalized the depth to one. This effects the bounds on q_i as well. Suppose a lower bound on t_3 for the original homogeneous camera is t_3^{low} , then the new bounds for the dehomogenized quaternions become $-1/t_3^{low} \leq q_i \leq 1/t_3^{low}$ for $i = 1, \dots, 4$. A conservative lower bound on t_3 can easily be obtained by examining the distances between the given world points.

3 Experiments

Even though the cost-function is highly non-convex, one might ask if local minima actually occur for realistic scenarios and if so, how often. Therefore, we first generated random 3D points within the unit cube $[-1, 1]^3$ and a random camera with viewing direction toward the origin, with a distance of two units. Then, the projected image coordinates were perturbed with independent Gaussian noise with different noise levels. To the left of Figure 3, a histogram

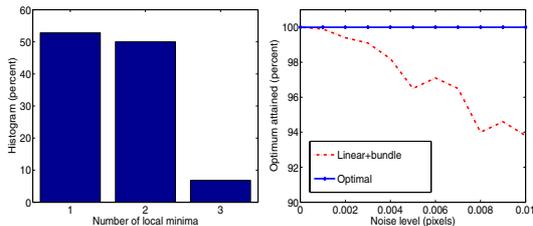


Figure 3. Left: Histogram of local minima with 4 points. Right: The percentage of times the global optimum is attained for 6 points.

of the number of local minima that occur for four points is plotted. The local minima have been computed with random initializations (in total, 100 tries) and the experiment has been repeated 1000 times. Note that all local minima

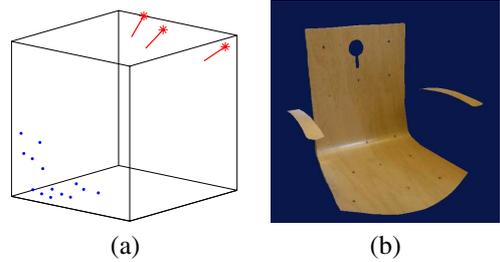


Figure 4. (a) Reconstructed cameras and (b) resulting VRML model

Residuals:	Our Alg.	Lin.Method	Lin.+Bundle
camera 1	1.351	10.76	1.351
camera 2	0.939	44.60	10.01
camera 3	0.950	4.741	0.950

Table 1. The RMS reprojection error measured in pixels, obtained when using all 14 points.

have positive depths (that is, world points are in front of the camera). There are typically 4-6 additional local minima with negative depths. To the right of Figure 3, the percentage of times the correct global optimum is reached for six points is shown. For our algorithm (optimal), the global optimum is of course always obtained. The traditional way is to apply a linear algorithm (DLT) which requires at least six points, and then do local refinements (bundle adjustment) [4]. As one can see, one might get trapped in a local minimum even for small noise levels. The following two subsections present experiments made with real data.

Chair Experiment

The setup for the chair experiment can be viewed in Figure 1. We used a MicroScribe-3DLX 3d scanner to measure the 3D-coordinates of the black points on the chair. For the first experiment we took three images of the chair and used the images of the 14 scanned 3D points to calculate the rotation and translation using our method. The intrinsic camera calibration was computed with standard techniques [4]. The reconstructed cameras are shown in Figure 4a. Using the images and the reconstructed cameras, it is easy to get a textured 3D model from the scanned model, see Figure 4b. Table 1 shows the resulting reprojection error measured in pixels. For this particular camera the resolution was 1360×2048 and the focal length was 1782. For comparison we also tried to solve the problem with a linear method with and without bundle adjustment. The linear method first calculates a projective 3×4 camera matrix P , and then use SVD factorization to find the closest camera matrix such that $\hat{P} = [R \mid t]$ where R is a scaled rotation matrix. As expected the linear method without bundle performs poorly. Also note that for the second chair image the bundle adjustment yields a local minimum. In Figure 5 the difference between the solutions obtained when not using all points is

illustrated. Here we regard the solution obtained when using all 14 points as the true solution. In Figure 5 the angles between the principal axis of the resulting camera matrix and the principal axis of the true solution are plotted. The red bars show the results when using from 4 to 13 points in the first image. The black bars show the same when using the second image, and the yellow bars give the result for the third image. Note that the computed solution is very similar already from 4 points to that of using all 14 points.

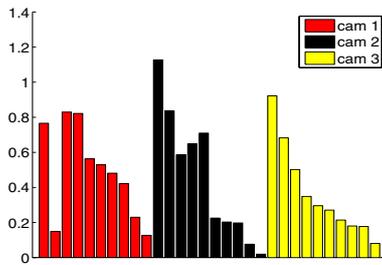


Figure 5. Angles in degrees between the principal axis of the optimal solution with 14 points and the solutions with 4-13 points, for the three chair experiments.

To illustrate the convergence of the algorithm, the performance of the algorithm for the two first images of the chair is plotted in Figure 6. To the left, the number of feasible rectangles at each iteration for the first chair image using 6 (red), 9 (green) and 12 (blue) points is given. To the right, the analogous plot for the second chair image is given.

Dinosaur Experiment

To further demonstrate the robustness of the algorithm, we have tested the algorithm on the publicly available turntable sequence of a dinosaur, see Figure 7b for one of the 36 images. The full reconstruction of 3D points and camera motion are also available, obtained by standard structure and motion algorithms [4]. For each of the 36 views, we have taken four randomly chosen points visible in that image and then estimated the camera pose. The resulting camera trajectory including viewing direction is compared to the original

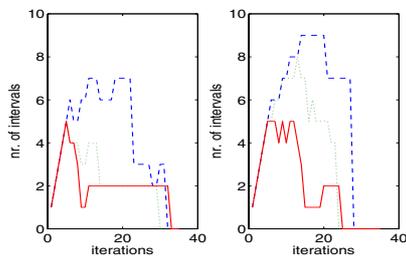


Figure 6. Convergence of the branch and bound algorithm. The number of feasible rectangles at each iteration for two of the chair images using 6 (red), 9 (green) and 12 (blue) points.

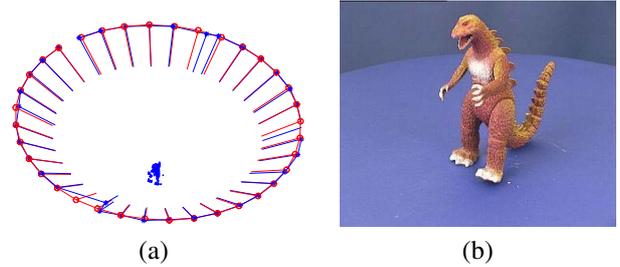


Figure 7. (a) The recovered camera motion for the dino experiment. Camera motion from full bundle adjustment *red curve* and using only four points *blue curve*. (b) One of 36 images of the dinosaur sequence.

camera motion in Figure 7a. Note the even though only four points have been used, the camera motion (*blue curve*) is very close to that of full bundle adjustment using all points (*red curve*).

4 Conclusions

In this paper, a globally optimal algorithm for perspective camera pose has been presented. The algorithm has been tested on both synthetic and real data, showing good performance. In addition, we have demonstrated that local minima do occur in practice making traditional algorithms less attractive.

References

- [1] S. Altmann. *Rotations, Quaternions and Double Groups*. Clarendon Press, 1986.
- [2] J. A. Grunert. Das pothenot'sche problem in erweiterter gestalt; nebst bemerkungen über seine anwendung in der geodäsie. *Grunert Archiv der Mathematik und Physik*, 1(3):238–248, 1841.
- [3] R. M. Haralick, C. N. Lee, K. Ottenberg, and M. Nolle. Review and analysis of solutions of the 3-point perspective pose estimation problem. *Int. Journal of Computer Vision*, 13(3):331–356, December 1994.
- [4] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. Second Edition.
- [5] F. Kahl and D. Henrion. Globally optimal estimates for geometric reconstruction problems. In *Int. Conf. Computer Vision*, pages 978–985, Beijing, China, 2005.
- [6] B. Kolman and R. E. Beck. *Elementary Linear Programming with Applications*. Academic Press, 1995.
- [7] C. Olsson, F. Kahl, and M. Oskarsson. The registration problem revisited: Optimal solutions from points, lines and planes. In *Conf. Computer Vision and Pattern Recognition*, New York City, USA, 2006. To appear.
- [8] L. Quan and Z. Lan. Linear $n \leq 4$ -point camera pose determination. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(8):774–780, August 1999.
- [9] H. S. Ryoo and N. V. Sahinidis. Analysis of bounds for multilinear functions. *Journal of Global Optimization*, 19:403–424, 2001.