A Global Optimal Solution to Non-Minimal Relative Pose Estimation

Sean Foley, Nathan Hatch, Apoorva Beedu

ECE 8823 Convex Optimization Final Project
April 2019

Relative Pose Estimation

Finding relative pose between two calibrated cameras is a fundamental problem in simultaneous localization and mapping (SLAM), visual odometry (VO) and structure from motion (SfM). *Relative pose estimation* of a moving camera consists of finding the camera pose (i.e. rotation and translation) with respect to the coordinate frame of a previous position. What is interesting in the context of ECE 8823 is that it can be formulated as an optimization problem and solved using convex relaxation.

Figure 1 shows the basic two-view camera setup. If p and p' are two image points looking at the same world point P, then the bearing vectors $f = \frac{p}{||p||}$ and $f' = \frac{p'}{||p'||}$ satisfy the *coplanarity constraint*

$$fEf' = 0, \tag{1}$$

where E is the *essential matrix* defined by

$$E = [t]_\times R$$

where $[t]_\times$ is the cross product matrix formed by vector t,

$$[t]_\times = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$$

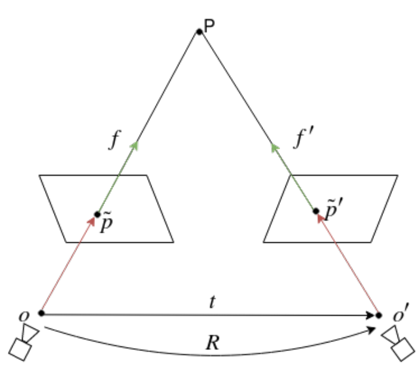


Figure 1: The relative pose problem. $\{f,\ f'\}$ are the bearing vectors.

Essential Matrix Manifold

We denote the essential matrix E as

$$E = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix}$$

and its corresponding vector

$$\boldsymbol{e} = \mathrm{vec}(E) = \begin{bmatrix} e_{11} & e_{21} & e_{31} & e_{12} & e_{22} & e_{32} & e_{13} & e_{23} & e_{33} \end{bmatrix}^T$$

Then the set of valid essential matrices

$$\mathcal{M}_E \triangleq \{E = [t]_\times R, \mid R \in \mathcal{SO}(3),\, t \in \mathcal{S}^2\}$$

is called the *normalised essential matrix manifold.* This turns out to be equal to the set [2]

$$\{E \mid EE^T = [t]_\times [t]_\times^T \text{ for some } t \text{ satisfying } t^T t = 1\}$$

Minimising the Error

Equation 1 holds for noise-free cases, but the equality will not strictly hold when measurement noise exists. Instead, [1] pursue the optimal pose by minimising the *algebraic error*

$$\min_{E \in \mathcal{M}_E} \sum_{i=1}^{\mathcal{N}} (f_i^T E f_i')^2. \tag{2}$$

By writing $\boldsymbol{f}_i = \mathrm{vec}(f_i' f_i^T)$ and observing that $f_i^T E f_i' = \boldsymbol{e}^T \boldsymbol{f}_i$, this objective can be reformulated as

$$\sum_{i=1}^{\mathcal{N}} (f_i^T E f_i')^2 = \boldsymbol{e}^T \left(\sum_{i=1}^{\mathcal{N}} \boldsymbol{f}_i \boldsymbol{f}_i^T \right) \boldsymbol{e} \triangleq \boldsymbol{e}^T \mathcal{C} \boldsymbol{e} \tag{3}$$

Note that $\mathcal{C}$ is positive semidefinite.

QCQP Formulation

The constraints $E \in \mathcal{M}_E$ are quadratic, and so is the objective (3), making this a *quadratically constrained quadratic program* (QCQP). More concretely, let $x \in \mathbb{R}^{12}$ be the concatenation of $\boldsymbol{e}$ and t. We can write (2) in the form

$$\min_x x^T C_0 x \quad \text{s.t.} \quad x^T A_i x = b_i \quad i = 1, \ldots, 7$$

where

$$C_0 = \begin{bmatrix} \mathcal{C} & 0 \\ 0 & 0_{3\times 3} \end{bmatrix}$$

and the matrices $A_i \in \mathbb{R}^{12\times 12}$ encode the constraints

$$\begin{aligned}
h_1 &= e_1^T e_1 - t_2^2 - t_3^2 = 0 \\
h_2 &= e_2^T e_2 - t_1^2 - t_3^2 = 0 \\
h_3 &= e_3^T e_3 - t_1^2 - t_2^2 = 0 \\
h_4 &= e_1^T e_2 + t_1 t_2 = 0 \\
h_5 &= e_1^T e_3 + t_1 t_3 = 0 \\
h_6 &= e_2^T e_3 + t_2 t_3 = 0 \\
h_7 &= t^T t = 1
\end{aligned}$$

Convex Relaxation

Unfortunately, QCQPs are in general NP-hard to solve [1]. As discussed in Professor Davenport's last lecture, one approach is to relax it to a semidefinite program (SDP) with variable $X = xx^T$:

$$\min_{X \in \mathbb{R}^{12\times 12}} \operatorname{trace}(C_0 X) \quad \text{s.t.} \quad X \succeq 0, \quad \operatorname{trace}(A_i X) = b_i \quad i = 1, \ldots, 7 \tag{4}$$

This is equivalent to the original problem if we impose the additional (non-convex) constraint that $\operatorname{rank}(X) = 1$.

How much does this relaxation cost us? A recent paper by Zhao [1] proves that for relative pose estimation with small enough measurement noise, it costs nothing.

Proof. We will give the proof for the noise-free case. In this case, there is a ground truth $\bar{R} \in \mathcal{SO}(3), \bar{t} \in \mathcal{S}^2$ such that $f_i^T[\bar{t}]_\times \bar{R} f_i = 0$ for all i.

Let $\bar{\boldsymbol{e}} = \text{vec}([\bar{t}]_\times \bar{R})$ and let $\bar{x}$ be the concatenation of $\bar{\boldsymbol{e}}$ and $\bar{t}$. I claim that $\bar{x}$ solves the original QCQP. Indeed, it is feasible by construction, and because there is no noise, it has an objective value of $\bar{\boldsymbol{e}}^T \mathcal{C} \bar{\boldsymbol{e}} = 0$. Since $C_0 \succeq 0$, this is the best possible objective value.

To prove that the relaxation is tight, it remains to show that 0 is also the optimal value of the relaxed problem (4). We can do this with Lagrangian duality. It turns out that the dual of (4) is

$$\max_{\nu \in \mathbb{R}^7} b^T \nu \quad \text{s.t.} \quad C_0 - \sum_{i=1}^{7} \nu_i A_i \succeq 0$$

where $b = (b_1, \ldots, b_7)$. Because the choice $\nu = 0$ is feasible and achieves objective value 0, we conclude that the relaxation is tight. □

Zhao [1] also shows that there is a *neighborhood* of positive semidefinite matrices near the noise-free matrix $\mathcal{C}$ such that the semidefinite relaxation remains tight. In other words, for small enough measurement noise, the relaxation still costs us nothing. The proof of this more general tightness result requires tools from algebraic geometry.

Recovering the Essential Matrix

Finally, how do we recover the optimal essential matrix E from the solution $X^\star$ of the SDP? Tightness guarantees that there exists a rank-1 solution to the SDP (namely $\bar{x}\bar{x}^T$), but if the SDP has more than one solution, that may not be the solution that we find.

Empirically, Zhao finds that $X^\star$ always has the form

$$\begin{bmatrix} x_e^\star x_e^{\star T} & \Delta \\ \Delta & x_t^\star x_t^{\star T} \end{bmatrix}$$

where $x_e^\star \in \mathbb{R}^9$ and $x_t^\star \in \mathbb{R}^3$. The cross terms Δ do not matter, since all of the matrices $C_0, A_1, \ldots, A_7$ have zeros in those locations. Therefore, concatenating $x_e^\star$ and $x_t^\star$ gives a solution to the QCQP, and we can recover E

from $x_e^\star$. The relative pose R, t can then be extracted from E using standard computer vision techniques.

It is an open question how to prove that $X^\star$ will always have that form, and in fact our own experimental results below seem to indicate that this is not always the case.

Experimental Verification

In order to show the effective tightness in the noisy case, we need to use experiments, as otherwise the exact error bound is unknown. Therefore, we try this method on synthetic data and see how it performs.

As in [1], we construct synthetic data. We take one pose at the origin, pointed along the z-axis. We take another pose, translated randomly away from the origin with a max distance of 2. Its orientation is generated via random Euler angles bounded between -0.5 and 0.5 radians. We select points at random between 4 and 8 distance from the origin, only keeping points that can be seen by both cameras, which we define as having a focal length of 800 pixels. These points can all be transformed into the image planes of the respective cameras since the camera parameters are known.

The matrix C can be constructed from the matching 2d points and the system solved via the proposed method. Noise is added to the points to analyze its effect on accuracy.

Solving

In practice, we use the CVXPY library to solve this problem. For semi-definite programs like ours, CVXPY defaults to the Splitting Conic Solver (SCS) [8]. This uses ADMM to solve the following problem:

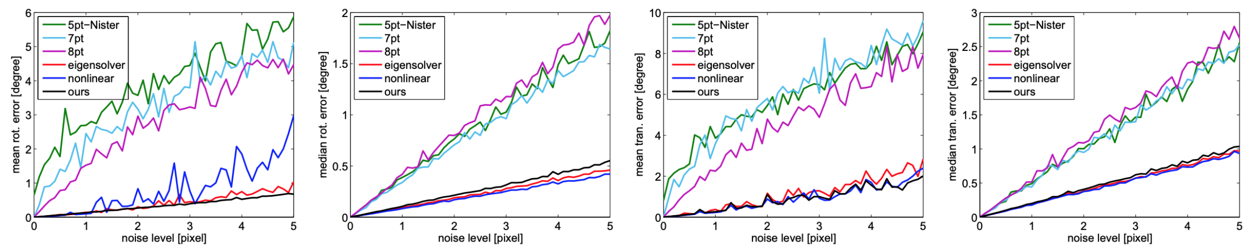
$$\begin{aligned} \text{minimize } & c^\top x \\ \text{subject to } & Ax + s = b \\ & s \in K \end{aligned}$$

Where K is a convex cone.

In [1] they use different solvers than we do, but SCS is able to find a solution to the problem as well, and extremely quickly.

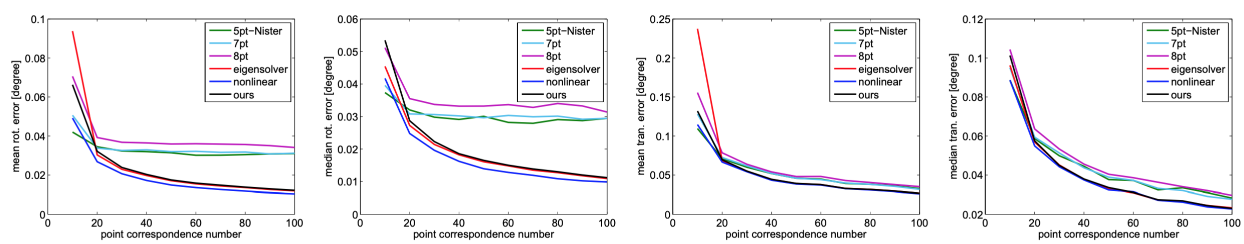
Results

In the noise-free case and noisy cases their solver performs as well as the best other method (eigensolver).



(a) mean ε_{rot} (b) median ε_{rot} (c) mean $\varepsilon_{\text{tran}}$ (d) median $\varepsilon_{\text{tran}}$

Figure 2: Results from [1]. Pose estimation accuracy with varying image noise level.



(a) mean ε_{rot} (b) median ε_{rot} (c) mean $\varepsilon_{\text{tran}}$ (d) median $\varepsilon_{\text{tran}}$

Figure 3: Results from [1]. Pose estimation accuracy with varying number of point correspondences.

However, in our case, we encountered a major problem - our solution X was consistently rank 5, and not rank 1, which prevents us from recovering the essential matrix. This is somewhat consistent with our theoretical findings, as we encountered some problems with their argument that the matrix would be rank 1. There exists a rank 1 solution, but it is not necessarily the one the solver will find. Perhaps the difference in our results is due to the difference in the solver we used, or perhaps there is some other subtlety.

References

[1] Zhao, J., 2019. An Efficient Solution to Non-Minimal Case Essential Matrix Estimation. arXiv preprint arXiv:1903.09067.

[2] Batra, D., Nabbe, B. and Hebert, M., 2007, February. An alternative formulation for five point relative pose problem. In 2007 IEEE Workshop on Motion and Video Computing (WMVC'07) (pp. 21-21). IEEE.

[3] J. Briales, L. Kneip, and J. Gonzalez-Jimenez. A Certifiably Globally Optimal Solution to the Non-minimal Relative Pose Problem. CVPR 2018.

[4] Y. Ding. On efficient semidefinite relaxations for quadratically constrained quadratic programming. 2007.

[5] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. IROS 2012.

[6] O. Faugeras. Three-dimensional Computer Vision: A Geometric Viewpoint. MIT Press, 1993.

[7] S. Boyd and L. Vandenberghe. Convex Optimization, 7th ed. Cambridge University Press, 2009.

[8] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd. Conic Optimization via Operator Splitting and Homogeneous Self-Dual Embedding. Journal of Optimization Theory and Applications, 2016.