Solving the Constrained Perspective-n-Point Problem

Joshua Nichols

March 18th 2025

1 Introduction

In this writeup we discuss a version of the perspective-n-point problem with the solution constrained to lie in the SE(2) group. This approach is useful in robotics where we wish to solve for the pose of a robot.

talk about opencv here

Note the notation and problem structure is heavily based on [1].

2 Problem Formulation

Suppose we have N image points $\mathbf{p}_i = [x_i, y_i]^\intercal$ and corresponding world points $\mathbf{P}_i = [X_i, Y_i, Z_i]^\intercal$. Let $K \in \mathbb{R}^{3\times3}$ be the camera's intrinsic matrix. The objective is to find the 3D pose, represented by the rotation matrix $R \in \mathbb{R}^{3\times3}$ and translation vector $T \in \mathbb{R}^3$, such that the reprojection error is minimized.

The additional constraint we introduce is to restrict the pose to lie in the Euclidian group $SE(2)$. With this constraint, the rotation matrix and translation vector can be written as

$$T = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} \text{ and } R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ where } x, y, \theta \in \mathbb{R}. \tag{1}$$

We use the Cayley transform to introduce a change of variables

$$R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1-\tau^2}{1+\tau^2} & \frac{2\tau}{1+\tau^2} & 0 \\ -\frac{2\tau}{1+\tau^2} & \frac{1-\tau^2}{1+\tau^2} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ where } \tau \in \mathbb{R}. \tag{2}$$

We assume the pinhole camera model, which models the relationship between image points and world points as

$$s \begin{bmatrix} \mathbf{p_i} \\ 1 \end{bmatrix} = K \left[R, T\right] \begin{bmatrix} \mathbf{P}_i \\ 1 \end{bmatrix}, \text{ for some } s \in \mathbb{R}. \tag{3}$$

To simplify the problem, we multiply both sides by K^{-1}

$$sK^{-1} \begin{bmatrix} \mathbf{p_i} \\ 1 \end{bmatrix} = \left[R, T\right] \begin{bmatrix} \mathbf{P}_i \\ 1 \end{bmatrix}, \text{ for some } s \in \mathbb{R}. \tag{4}$$

For the rest of the paper, we will use normalized image points $\mathbf{u_i} = [u_i, v_i]^\intercal$ defined as $\begin{bmatrix} \mathbf{u}_i \\ 1 \end{bmatrix} = K^{-1} \begin{bmatrix} \mathbf{p_i} \\ 1 \end{bmatrix}$. Rewriting (4) with normalized image points gives

$$s \begin{bmatrix} \mathbf{u_i} \\ 1 \end{bmatrix} = \begin{bmatrix} R, T \end{bmatrix} \begin{bmatrix} \mathbf{P}_i \\ 1 \end{bmatrix}, \text{ for some } s \in \mathbb{R}. \tag{5}$$

For notational purposes, let $\mathbf{r_i}$ represent the i-th row of the rotation matrix and t_i represent the i-th element of the translation vector. From (5) we can solve for s

$$\begin{bmatrix} su_i \\ sv_i \\ s \end{bmatrix} = \begin{bmatrix} R, T \end{bmatrix} \begin{bmatrix} \mathbf{P_i} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1\mathbf{P}_i + t_1 \\ \mathbf{r}_2\mathbf{P}_i + t_2 \\ \mathbf{r}_3\mathbf{P}_i + t_3 \end{bmatrix} \implies s = \mathbf{r}_3\mathbf{P}_i + t_3. \tag{6}$$

Substituting $s = \mathbf{r}_3\mathbf{P}_i + t_3$ into (5) based on the result from (6) gives

$$u_i\big(\mathbf{r_3}\mathbf{P}_i + t_3\big) - \mathbf{r_1}\mathbf{P}_i + t_1 = 0, \tag{7}$$
$$v_i\big(\mathbf{r_3}\mathbf{P}_i + t_3\big) - \mathbf{r_2}\mathbf{P}_i + t_2 = 0. \tag{8}$$

Substituting the definition of R into (7) and (8) simplifies to

$$u_iZ_i(1+\tau^2) - X_i(1-\tau^2) - Y_i(2\tau) - t_1(1+\tau^2) = 0, \tag{9}$$
$$v_iZ_i(1+\tau^2) + X_i(2\tau) - Y_i(1-\tau^2) - t_2(1+\tau^2) = 0. \tag{10}$$

Next, we introduce another change of variables

$$t_i' = t_i(1+\tau^2). \tag{11}$$

Substituting (10) into (8) and (9) gives

$$E_i^x = u_iZ_i(1+\tau^2) - X_i(1-\tau^2) - Y_i(2\tau) - t_1' = 0, \tag{12}$$
$$E_i^y = v_iZ_i(1+\tau^2) + X_i(2\tau) - Y_i(1-\tau^2) - t_2' = 0. \tag{13}$$

This reduces the problem such that (12) and (13) are now both second degree polynomials. Factoring out τ gives

$$E_i^x = (u_i Z_i + X_i)\tau^2 - 2Y_i\tau + u_i Z_i - X_i - t_1' = 0, \tag{14}$$
$$E_i^y = (v_i Z_i + Y_i)\tau^2 + 2X_i\tau + v_i Z_i - Y_i - t_2' = 0. \tag{15}$$

We can now formulate this as a least-squares problem

$$\min_{t_1', t_2', \tau} \sum_{i=1}^{N} (E_i^x)^2 + (E_i^y)^2. \tag{16}$$

To further simplify the problem, we can find a closed form solution for t_1' and t_2' in terms of τ. If we fix τ as a constant, we have the following optimization problem

$$\min_{t_1', t_2'} \sum_{i=1}^{N} (E_i^x)^2 + (E_i^y)^2 = \min_{t_1'} \sum_{i=1}^{N} (E_i^x)^2 + \min_{t_2'} \sum_{i=1}^{N} (E_i^y)^2. \tag{17}$$

First we solve for t_1'. The first order optimality condition is

$$\frac{d}{dt_1'} \sum_{i=1}^{N} (E_i^x)^2 = 0 \implies \sum_{i=1}^{N} \left(\frac{d}{dt_1'} (E_i^x)^2 \right) = 0 \tag{18}$$

Differentiating by the chain rule gives

$$\frac{d}{dt_1'} (E_i^x)^2 = 2 \left(\frac{d}{dt_1'} E_i \right) E_i^x = -2E_i^x. \tag{19}$$

Substituting (19) into (18)

$$\sum_{i=1}^{N} \left(-2E_i^x \right) = \sum_{i=1}^{N} E_i^x = 0. \tag{20}$$

Finally solving for t_1' gives

$$t_1' = \frac{1}{N}\sum_{i=1}^{N}\Big((u_iZ_i + X_i)\tau^2 - 2Y_i\tau + u_iZ_i - X_i\Big). \tag{21}$$

Repeating the same for t_2' gives

$$t_2' = \frac{1}{N}\sum_{i=1}^{N}\Big((v_iZ_i + Y_i)\tau^2 + 2X_i\tau + v_iZ_i - Y_i\Big). \tag{22}$$

Observe t_1' and t_2' are second degree polynomials of τ. Let us define the constants

$$A_1 = \frac{1}{N}\sum_{i=1}^{N}(u_iZ_i + X_i) \tag{23}$$

$$B_1 = \frac{1}{N}\sum_{i=1}^{N}(-2Y_i) \tag{24}$$

$$C_1 = \frac{1}{N}\sum_{i=1}^{N}(u_iZ_i - X_i) \tag{25}$$

$$A_2 = \frac{1}{N}\sum_{i=1}^{N}(v_iZ_i + Y_i) \tag{26}$$

$$B_2 = \frac{1}{N}\sum_{i=1}^{N}(2X_i) \tag{27}$$

$$C_2 = \frac{1}{N}\sum_{i=1}^{N}(v_iZ_i - Y_i) \tag{28}$$

Now we can rewrite the errors as

$$E_i^x = (u_iZ_i + X_i - A_1)\tau^2 + (-2Y_i - B_1)\tau + u_iZ_i - X_i - C_1 = 0, \tag{29}$$

$$E_i^y = (v_iZ_i + Y_i - A_2)\tau^2 + (2X_i - B_2)\tau + v_iZ_i - Y_i - C_2 = 0. \tag{30}$$

Substituting back into our least-squares problem, we now have a single variable optimization problem

$$\min_{\tau} C(\tau), \qquad C(t) = \sum_{i=1}^{N} (E_i^x)^2 + (E_i^y)^2. \tag{31}$$

Observe $C(\tau)$ is a fourth degree polynomial. Thus the first-order optimality condition is simply a third-degree polynomial of τ. Cardano's formula provides a closed form solution to this polynomial.

The final step is to recover t_1, t_2, and θ as

$$t_1 = \frac{t_1'}{(1+\tau^2)}, \tag{32}$$

$$t_2 = \frac{t_2'}{(1+\tau^2)}, \tag{33}$$

$$\theta = 2\tan^{-1}(\tau). \tag{34}$$

Thus we have shown a closed form solution for the constrained perspective-n-point problem.

References

[1] F. Zhou and S. Coauthor, "An efficient and accurate algorithm for the perspecitve-n-point problem," https://www.cs.cmu.edu/~kaess/pub/Zhou19iros.pdf.