Closed-form Solution for Inverse-depth Feature Marginalization for Robust two, three views Bundle Adjustments

Tien Do, Leo Neira and Stergios I. Roumeliotis
Department of Computer Science & Engineering
University of Minnesota

MARS LAB

Multiple Autonomous
Robotic Systems Laboratory

Technical Report
Number 2019-001
May 2019

Dept. of Computer Science & Engineering
University of Minnesota
4-192 EE/CS Building
200 Union St. S.E.
Minneapolis, MN 55455
Tel: (612) 625-2217
Fax: (612) 625-0572
URL: `http://mars.cs.umn.edu/`

Closed-form Solution for Inverse-depth Feature Marginalization for Robust two, three views Bundle Adjustments

Tien Do, Leo Neira and Stergios I. Roumeliotis
Department of Computer Science & Engineering
University of Minnesota

Multiple Autonomous Robotic Systems Laboratory, TR-2019-001
May 2019

Abstract

In this technical report, we briefly describe the relationship between the 2pt [1], P3P [2] and 5pt [3] solvers; and the efficient solver for $N(=2,3)$ views bundle adjustments (BA) in the square root information domain.

1 The 2pt/P3P/5pt solvers

Without loss of generality, we will use $\{C_1\},\{C_2\},\{C_3\}$ as a generic frames, indicating 3 distinct views, without referring explicitly to any of $\{G\},\{C_k\},\{M\}$. The geometric constraint between 2 cameras C_1 and C_2 viewing the same feature $\mathbf{f}_i$ is described as:

$$\begin{bmatrix} {}^{C_1}\mathbf{f}_i \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}({}^{C_1}\mathbf{q}_{C_2}) & {}^{C_1}\mathbf{t}_{C_2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{C_2}\mathbf{f}_i \\ 1 \end{bmatrix} \tag{1}$$

where ${}^{C_1}\mathbf{f}$ and ${}^{C_2}\mathbf{f}$ are the 3D feature positions with respect to C_1 and C_2. When 3 pairs of 3D-2D correspondences satisfying eq. (1) are given, i.e., ${}^{C_1}\mathbf{f}$ and its corresponding bearing vector ${}^{C_2}\mathbf{b}_{f_i} = \frac{{}^{C_2}\mathbf{f}_i}{||{}^{C_2}\mathbf{f}_i||}$ in the second image, we employ the P3P RANSAC to determine ${}^{C_1}\mathbf{R}_{C_2}$ and ${}^{C_1}\mathbf{t}_{C_2}$.

When only 2D-2D correspondences are available, satisfying eq. (1) are given, we can form the well-known epipolar constraints:

$${}^{C_1}\mathbf{b}_{f_i}^T R({}^{C_1}\mathbf{q}_{C_2}) \lfloor {}^{C_1}\mathbf{t}_{C_2} \times \rfloor {}^{C_2}\mathbf{b}_{f_i} = 0 \tag{2}$$

Then, given 5 pairs of points, we can obtain ${}^{C_1}\mathbf{R}_{C_2}$ and ${}^{C_1}\mathbf{t}_{C_2}$ up to scale using the 5pt RANSAC.

When the baseline is small, i.e. ${}^{C_1}\mathbf{t}_{C_2} \approx 0$, we can approximate eq. (1) as

$${}^{C_1}\mathbf{f}_i = R(\delta\mathbf{q} \otimes {}^{C_1}\mathbf{q}_{C_2}) {}^{C_2}\mathbf{f}_i \tag{3}$$

where translation is considered as rotational noise. Given 2 pairs of correspondences satisfying eq. (3), we can compute ${}^{C_1}\mathbf{q}_{C_2}$ using the 2 pairs of 2pt RANSAC.

2 Inverse-depth robust 2-view BA

After ${}^{C_1}\mathbf{R}_{C_2}$ and ${}^{C_1}\mathbf{t}_{C_2}$ are found, we follow the outline of [1] to perform 2-view reconstruction. We begin by representing the features with the inverse-depth parameterization such that

$${}^{C}\mathbf{f}_i = \frac{1}{\lambda_i} \begin{bmatrix} \alpha_i \\ \beta_i \\ 1 \end{bmatrix} \tag{4}$$

Additionally, given that we have the camera intrinsic parameters, we define ${}^{C}{}_i\mathbf{z}$ as homogeneous coordinate of the 3D feature i in image C_k Next, we seek to find the optimal solution

$$\mathbf{y} = \begin{bmatrix} {}^{C_2}\mathbf{q}_{C_1}^T & {}^{C_2}\mathbf{t}_{C_1}^T & {}^{C_1}\begin{bmatrix}\alpha_i & \beta_i & \lambda_i\end{bmatrix}_{i=1,2,\ldots,n} \end{bmatrix}^T \tag{5}$$

that minimizes the total reprojection error:

$$
\begin{aligned}
\mathbb{C}(\mathbf{y}) &= \sum_{i=1}^{n} \left(\rho(\|{}^{C_1}\mathbf{z}_i - \Pi({}^{C_1}\mathbf{f}_i)\|) + \rho(\|{}^{C_2}\mathbf{z}_i - \Pi({}^{C_1}\mathbf{f}_i)\|)\right) && (6)\\
&= \sum_{i=1}^{n} \left(\rho(\left\| {}^{C_1}\mathbf{z}_i - \begin{bmatrix} {}^{C_1}\alpha_i \\ {}^{C_1}\beta_i \end{bmatrix} \right\|) + \rho(\left\| {}^{C_2}\mathbf{z}_i - \Pi\left({}^{C_2}\mathbf{R}_{C_1} \begin{bmatrix} {}^{C_1}\alpha_i \\ {}^{C_1}\beta_i \\ 1 \end{bmatrix} + \lambda_i {}^{C_2}\mathbf{t}_{C_1} \right) \right\|)\right) && (7)\\
&= \sum_{i=1}^{n} \left(\rho({}^{C_1}e_i) + \rho({}^{C_2}e_i)\right) && (8)
\end{aligned}
$$

where $\rho(e)$ is the Huber robust cost function

$$
\rho(e) = \begin{cases} \frac{1}{2}e^2, & e \leq \sigma_p \\ \sigma_p|e| - \frac{1}{2}e^2, & e > \sigma_p \end{cases} \tag{9}
$$

The jacobian corresponding to each feature is:

$$
\mathbf{J}_{f_i} = \begin{bmatrix} s_1\mathbf{I} & \mathbf{0} \\ s_2\mathbf{J}_{f_{i_1}} & s_2\mathbf{J}_{f_{i_2}} \end{bmatrix} \tag{10}
$$

where

$$
\begin{aligned}
\mathbf{J}_{f_{i_1}} &= {}^{C_2}\mathbf{R}_{C_1(1:2,1:2)} - \frac{1}{{}^{C_2}z_i} \begin{bmatrix} {}^{C_2}x_i \\ {}^{C_2}y_i \end{bmatrix} {}^{C_2}\mathbf{R}_{C_1(3,1:2)} \\
\mathbf{J}_{f_{i_2}} &= {}^{C_2}\mathbf{t}_{C_1(1:2)} - \frac{{}^{C_2}\mathbf{t}_{C_1(3)}}{{}^{C_2}z_i} \begin{bmatrix} {}^{C_2}x_i \\ {}^{C_2}y_i \end{bmatrix} \\
\begin{bmatrix} {}^{C_2}x_i \\ {}^{C_2}y_i \\ {}^{C_2}z_i \end{bmatrix} &= {}^{C_2}\mathbf{R}_{C_1} \begin{bmatrix} \alpha_i \\ \beta_i \\ 1 \end{bmatrix} + \lambda_i {}^{C_2}\mathbf{t}_{C_1}
\end{aligned}
$$

and s_1 and s_2 are computed as weighting factors as:

$$
s_j = \begin{cases} 1, & {}^{C_j}e_i \leq \sigma_p \\ \sqrt{\frac{\sigma_p}{|{}^{C_j}e_i|}}, & {}^{C_j}e_i > \sigma_p \end{cases} \tag{11}
$$

The analytical closed form solution of the left null space of each feature's Jacobian is:

$$
\mathbf{u}_{f_i}^T = normalize(\left[\begin{bmatrix} -\mathbf{J}_{f_{i_1}(2)} & \mathbf{J}_{f_{i_2}(1)} \end{bmatrix} s_2\mathbf{J}_{f_{i_1}} \quad \begin{bmatrix} -s_1\mathbf{J}_{f_{i_2}(2)} & s_1\mathbf{J}_{f_{i_2}(1)} \end{bmatrix}\right]) \tag{12}
$$

and the Jacobian of the pose for each feature is defined as

$$
\mathbf{J}_{r_i}^{(2)} = \mathbf{J}_{\Pi}^{(2)} \begin{bmatrix} \lfloor {}^{C_2}\mathbf{R}_{C_1} \begin{bmatrix} \alpha_i \\ \beta_i \\ 1 \end{bmatrix} \times \rfloor & -\lambda_i \hat{\mathbf{t}}^{\perp\perp} & \lambda_i \hat{\mathbf{t}}^{\perp} \end{bmatrix} \tag{13}
$$

where

$$
\mathbf{J}_{\Pi}^{(2)} = \mathbf{J}_{\Pi}(\begin{bmatrix} {}^{C_2}x_i \\ {}^{C_2}y_i \\ {}^{C_2}z_i \end{bmatrix}) = \frac{1}{{}^{C_2}z_i} \begin{bmatrix} 1 & 0 & -\frac{{}^{C_2}x_i}{{}^{C_2}z_i} \\ 0 & 1 & -\frac{{}^{C_2}y_i}{{}^{C_2}z_i} \end{bmatrix} \tag{14}
$$

We then apply Gauss-Newton with $\mathbf{J}_{f_i}$, $\mathbf{J}_{r_i}^{(2)}$, and $\mathbf{u}_{f_i}$ as described in [1]

3 Inverse-depth robust 3-view BA

Once there exists a After getting the results from P3P, we can refine the estimate by employing a 3-view BA. To do so, we extend the cost function given in Eqn. 6 to include the third view. We now seek to optimize over the vector

$$\mathbf{y} = \begin{bmatrix} {}^{C_2}\mathbf{q}_{C_1}^T & {}^{C_2}\mathbf{t}_{C_1}^T & {}^{C_3}\mathbf{q}_{C_1}^T & {}^{C_3}\mathbf{p}_{C_1}^T & {}^{C_1}\begin{bmatrix}\alpha_i & \beta_i & \lambda_i\end{bmatrix}_{i=1,2,\ldots,n}\end{bmatrix}^T \tag{15}$$

where ${}^{C_3}\mathbf{p}_{C_1}$ is the translation between C_3 and C_1 with scale defined by the first 2 views. The updated cost function is given by:

$$\mathbb{C}(\mathbf{y}) = \sum_{i=1}^{n} \left(\rho({}^{C_1}e_i) + \rho({}^{C_2}e_i) + \rho({}^{C_3}e_i)\right) \tag{16}$$

with

$${}^{C_3}e_i = \left\| {}^{C_3}\mathbf{z}_i - \Pi\left({}^{C_3}\mathbf{R}_{C_1} \begin{bmatrix}\alpha_i \\ \beta_i \\ 1\end{bmatrix} + \lambda_i {}^{C_3}\mathbf{p}_{C_1}\right)\right\| \tag{17}$$

From the cost function, we compute the following Jacobians similarly to the previous section:

$$\mathbf{J}_{f_i} = \begin{bmatrix} s_1\mathbf{I} & \mathbf{0} \\ s_2\mathbf{J}_{f_{i_1}}^{(2)} & s_2\mathbf{J}_{f_{i_2}}^{(2)} \\ s_3\mathbf{J}_{f_{i_1}}^{(3)} & s_3\mathbf{J}_{f_{i_2}}^{(3)} \end{bmatrix} \tag{18}$$

$$\mathbf{J}_{r_i} = \begin{bmatrix} \mathbf{J}_{r_i}^{(2)} & \begin{bmatrix} \mathbf{J}_{\Pi}^{(3)} & \lfloor {}^{C_3}\mathbf{R}_{C_1} \begin{bmatrix}\alpha_i \\ \beta_i \\ 1\end{bmatrix} \times\rfloor & \lambda_i \mathbf{J}_{\Pi}^{(3)} \end{bmatrix} \end{bmatrix} \tag{19}$$

$$\mathbf{J}_{\Pi}^{(3)} = \mathbf{J}_{\Pi}({}^{C_3}\mathbf{R}_{C_1} \begin{bmatrix}\alpha_i \\ \beta_i \\ 1\end{bmatrix} + \lambda_i {}^{C_3}\mathbf{p}_{C_1}) \tag{20}$$

However, the left null space is now become a bit more complicated:

$$\mathbf{U}_{f_i}^T = \begin{bmatrix} -\begin{bmatrix}\mathbf{u_1}_i^T & \mathbf{u_2}_i^T\end{bmatrix} \begin{bmatrix} s_2\mathbf{J}_{f_{i_1}}^{(2)} \\ s_3\mathbf{J}_{f_{i_1}}^{(3)} \end{bmatrix} & s_1\mathbf{u}_{1_i}^T & s_1\mathbf{u}_{2_i}^T \end{bmatrix} \tag{21}$$

where

$$\begin{bmatrix}\mathbf{u_1}_i^T & \mathbf{u_2}_i^T\end{bmatrix} = \begin{bmatrix} \mathbf{J}_{f_{i_1}(2)}^{(2)} & -\mathbf{J}_{f_{i_1}(1)}^{(2)} & 0 & 0 \\ s_3\mathbf{J}_{f_{i_1}(1)}^{(3)} & 0 & -s_2\mathbf{J}_{f_{i_1}(1)}^{(2)} & 0 \\ s_3\mathbf{J}_{f_{i_1}(2)}^{(3)} & 0 & 0 & -s_2\mathbf{J}_{f_{i_1}(1)}^{(1)} \end{bmatrix} \tag{22}$$

Before executing the algorithm in [1], we will first need to find the orthonormal basis of the row space spanned in $\mathbf{U}_{f_i}^T$. This can simply obtained through a Gram-Schmidt process.

References

[1] T. Do, L. C. Carrillo-Arce, and S. I. Roumeliotis. "High-speed autonomous quadrotor navigation through visual and inertial paths". *The International Journal of Robotics Research*, 38(4):486–504, Aug. 2019.

[2] T. Ke and S. Roumeliotis. "An efficient algebraic solution to the perspective-three-point problem". In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7225–7233, Honolulu, HI, Jul. 21–26 2017.

[3] H. Stewenius, C. Engels, and D. Nistér. "Recent developments on direct relative orientation". *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(4):284–294, Jun. 2006.