Solving Perspective-n-Point Problem with Spherical Regression

Ying He, Suilao Li and Qiang Guo
Northwestern Polytechnical University, College of Automation, 710129 Xi’an, China
cnheying@qq.com, lisuilao@nwpu.edu.cn, guofeng422@163.com

Abstract. This paper proposes a spherical regression relaxation solution for perspective-n-point problem. The problem is formulated as a minimization problem using angle error based spherical cost. With the relaxation of scale constraint, translation parameters can be calculated in closed form. Then, rotation parameters are immediately solved in closed form using spherical regression. The scale, translation and rotation parameters are alternatively estimated while keeping the others fixed until convergence. The proposed method is simple and able to cope with arbitrary central camera models. Experiment results show that the proposed method achieves accuracy almost the same as the maximum-likelihood estimator, and its computational efficiency is even comparable with some state-of-the-art non-iterative methods.

1. Introduction

Perspective-n-Point (PnP) is the problem of determining the position and orientation of a calibrated camera given 3D points and their corresponding 2D projections. It is widely used in computer vision, photogrammetry, robotics, and navigation[1].

The minimal PnP problem, P3P, can be solved in closed form[2-4]. It is desirable to consider larger point sets to reduce the sensitivity to noise. The existing solutions for n ≥ 4 can be classified as non-iterative, semi-iterative and full-iterative methods.

Non-iterative methods solved PnP in closed form by using some approximate cost function. Li et al. [5] computed solutions from the stationary points of an algebraic cost function directly (RPnP). Hesch et al. [6] calculated the stationary points of the object space cost using an algebraic geometry solver (DLS). Zheng et al. [7] reduced the object space cost to 2 dimensions (OPnP). Zheng et al. [8] also used the 2D object space cost but divided the parameters space of quaternion into four independent subspaces (ASPnP). Wang et al. used the 2D object space cost to refine the solution of RPnP (SRPnP) [9]. However, from the perspective of the cost function, the object space cost implicitly assigns more weight to the reference points that are far from the camera, that causes biased solutions.

Full-iterative methods treated PnP as a nonlinear least square problem. The default algorithm in OpenCV minimized the reprojection cost with the Levenberg-Marquardt (LM) algorithm. Urban et al. [10] projected the reprojection error to the corresponding tangent space on the unit sphere and then minimized the projected error cost using the Gauss-Newton algorithm (MLPnP). Generally speaking, full-iterative algorithms have two drawbacks, 1) they may get stuck in an inappropriate local minimum if without appropriate initialization. 2) the complex optimization over full parameters lead to computational inefficiency.

To decrease the number of parameters involved in the optimization iteration, semi-iterative methods eliminated some parameters that can be calculated in closed form in advance. Lu et al. [11] eliminated translation and therefore its optimization involved rotation only (LHM). Garro et al. [12] formulated the PnP as the anisotropic orthogonal Procrustes problem to calculate rotation and scales.
(PPnP). However, to our best knowledge, all existing semi-iterative solutions minimize the object space cost thereby giving the aforementioned biased solutions.

This paper proposes a semi-iterative algorithm, which minimizes the sum of squared angle error, or spherical cost, and is able to cope with arbitrary central camera models. This is achieved based on the following facts: 1) As a camera is essentially an angle measurement device, angle error based cost is more reasonable. 2) with the spherical cost, the rotation can be efficiently solved in closed form with the spherical regression. 3) the spherical regression works with 3D direction vectors, thus it is able to cope with arbitrary central camera models.

We will show that the proposed method is better than the non-iterative state-of-the-art methods in terms of accuracy, and is the fastest iterative algorithm, even comparable with some state-of-the-art non-iterative methods.

In the remainder of the paper, the formulation of PnP and the spherical regression model is introduced in section II. The calculation of the optimal translation using the relaxation method and the optimal rotation using the spherical regression, is described in section III. Results of the experiment are given in section IV. In section V, we draw conclusions.

2. Mathematical Formulation

2.1. The PnP Problem

The Perspective-n-Point problem is to compute a rotation \( R \) and a translation \( t \) that map the \( n \) reference points \( p_i \) in the world frame to their corresponding direction unit-vectors \( q_i \) in the camera frame:

\[
\lambda_i q_i = R(p_i + t), i = 1..n
\]

where \( \lambda_i = \|p_i + t\| \) are the length of \( p_i \) in the camera frame.

2.2. The Spherical Regression

The spherical regression model hypothesizes \( N \) fixed unit vectors \( x_i \) and independent random variables \( y_i \) as a distribution with mean \( Rx_i \), where \( R \) is the unknown orthogonal matrix.

Assume that the measurements come from the Von Mises-Fisher distribution with spherical means \( Rx_i \):

\[
y_i \sim \kappa \exp(\kappa y_i^T Rx_i) / \sinh \kappa
\]

where \( \kappa \) is the concentrate parameter of the Von Mises-Fisher distribution.

With independent measurements, the likelihood function for \( R \) is:

\[
\ln L(R) \sim \kappa \text{Trace}(Y^T RX)
\]

where \( Y = (y_1^T, y_2^T, ..., y_N^T)^T \), \( X = (x_1^T, x_2^T, ..., x_N^T)^T \).

The spherical regression of the Von Mises-Fisher distribution says that[13]:

**Theorem 1** if \( R \in O(n) \) or \( \det(UV^T) = 1 \) then the unique maximization likelihood estimate of \( R \) is \( \hat{R}_{\text{sl}} = UV^T \). Otherwise, \( \hat{R}_{\text{sl}} = \text{Vdiag}(1,1,...,1)U^T \), which is unique if the smallest singular value of \( A \) is nondegenerate, where \( U\Sigma V = \text{svd}(A), A = XY^T \).

3. Optimal Rotation and Translation

The optimal rotation \( R' \) and translation \( t' \) are found by minimizing the sum of squared residuals \( r_i \) (\( i = 1..n \)).
\[ \{R^*, t^*\} = \arg \min_{R \in SO(3)} \sum_{i=1}^{n} \|v_i^* - \sum_{i=1}^{n} R^T q_i - \lambda_i^{-1} (p_i + t) \| \]  

(4)

Relaxing the scale constraints \( \lambda_i \) as a free parameter [6], the optimal translation parameters can be calculated immediately:

\[ t = \left( \sum_{i=1}^{n} \lambda_i^{-2} \right)^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-1} \left( R^T q_i - \lambda_i^{-1} p_i \right) \right). \]  

(5)

Combining 错误！未找到引用源。 and 错误！未找到引用源。, we obtain:

\[ \sum_{i=1}^{n} \| v_i^* \| = \sum_{i=1}^{n} \| R^T m_i - n_i \| \]  

(6)

where \( m_i = q_i - \lambda_i^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-2} \right)^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-1} q_i \right) \), \( n_i = \lambda_i^{-1} p_i - \lambda_i^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-2} \right)^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-2} p_i \right) \).

错误！未找到引用源。 can be reformulated as

\[ \{R^*\} = \arg \max_R \text{Trace}(M^T RN) \]  

(7)

where \( N = (n_1^T, n_2^T, ..., n_n^T)^T \) and \( M = (m_1^T, m_2^T, ..., m_n^T)^T \).

错误！未找到引用源。 is of the form of 错误！未找到引用源。 and only depends on \( \lambda_i \). With fixed \( \lambda_i \), the PnP problem is reduced to the spherical regression of the Von Mises-Fisher distribution.

Now, unknowns \( \lambda_i \), \( t \) and \( R \) are estimated iteratively in such a way, like in the LHM and PPnP, that each variable is alternatively calculated while keeping the others fixed. The algorithm can be summarized as follows:

**Algorithm 1. Spherical Perspective-n-Point (SPnP)**

1. \( \lambda_i = 1 \).
2. \( R = U \text{diag}(1,1,\det(UV^T))V^T \), \( U S V^T = \sum_{i=1}^{n} m_i n_i^T \),
   \[ m_i = q_i - \lambda_i^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-2} \right)^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-1} q_i \right) \],
   \[ n_i = \lambda_i^{-1} p_i - \lambda_i^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-2} \right)^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-2} p_i \right) \].
3. \( t = \left( \sum_{i=1}^{n} \lambda_i^{-2} \right)^{-1} \left( \sum_{i=1}^{n} \lambda_i^{-1} \left( R^T q_i - \lambda_i^{-1} p_i \right) \right). \)
4. \( \lambda_i = \| p_i + t \| \).
5. Iterate over steps 2,3,4 until convergence.

One might recognize in this alternation between computing pose and distance a resemblance of the PPnP algorithm.

4. Experiment Results
We compare the proposed algorithm with the following algorithms in terms of accuracy and execution time: 1) RPnP[5], a non-iterative method using an algebraic cost. 2) MLPnP[10], a iterative method using the reprojection error projected to the tangent space of the unit sphere. 3) EPnP+GN[14], a non-iterative algorithm using an algebraic cost with iterative Gauss-Newton refine. 4) OPnP[7], ASPnP[8], and SRPnP[9], all of them are non-iterative algorithms using the 2D object space error. 5) PPnP[12] and LHM[11] are semi-iterative method using the object space cost. 6) BA[15], traditional bundle adjustment using the reprojection cost.

All the codes of the chosen algorithms are publicly available codes from their paper’s author. We do not modify these codes except feeding and collecting data. For the sake of fairness, when comparing the accuracy and execution time, all algorithms that need initialization are initialized using the DLT[16] algorithm.

A virtual camera with a focal length 800 pixels and image resolution 640x480 pixels is used. The 3D reference points are randomly generated in the camera frame in the range of \([-2, 2] \times [-2, 2] \times [4, 8]\).

4.1. Translation and Rotation Errors

Fig. 1 shows the mean rotation error and position error against the number of reference points changes from 100 to 1000. The translation and rotation errors are defined as \(e_t = \|\mathbf{t} - \mathbf{t}_{\text{true}}\|\) and \(e_R = \|\log(\mathbf{R}/\mathbf{R}_{\text{true}})\|_F\), respectively, where \(\|\|_F\) is Frobenius norm. For each number of reference points, the experiment is repeated 1,000 times independently.

As shown in Fig. 1, both the mean rotation error and position error decrease as the number of reference points increase. LHM and PPnP overlap OPnP because they all use the object space error. The smallest error of BA verifies that the reprojection error is the gold standard. The overlap of SPnP and BA proves the similarity of the spherical error and the reprojection error.

4.2. Execution Time

Fig. 2 shows the execution time as the number of reference points increases from 100 to 1000 on a laptop with Windows 7, Intel i7 4720HQ 2.6G, 8G RAM, and Matlab R2018a.

As shown in Fig. 2, EPnP, ASPnP and RPnP run fastest because of the algebraic error. SPnP comes after RPnP that makes SPnP the fastest iterative algorithm. This could be due to the effectiveness of the spherical regression and the simplicity of SPnP.
5. Conclusion
The contribution of this paper is both theoretical and practical. On the theoretical side, it establishes the theoretical connection between spherical regression and perspective-n-point problem, which leads to the relaxation method. The spherical regression’s ability to work with 3D direction vectors enables the proposed method to cope with arbitrary central camera models. The similarity between the spherical error and the reprojection error makes the accuracy of the proposed method almost the same as the maximum-likelihood estimator. On the practical side, the effectiveness of the spherical regression and the simplicity of the proposed method result in high computational efficiency. The proposed method can be used for camera pose determination in computer vision, photogrammetry, robotics, virtual reality and navigation. In future work, we will try to generalize the proposed method to anisotropic noise case.

6. References
[1] Luo, X., Mori, K., Peters, T.M.: Advanced Endoscopic Navigation: Surgical Big Data, Methodology, and Applications. Annual Review of Biomedical Engineering 20, 221-251 (2018)
[2] Gao, X.-S., Hou, X.-R., Tang, J., Cheng, H.-F.: Complete solution classification for the perspective-three-point problem. Pattern Analysis and Machine Intelligence, IEEE Transactions on 25, 930-943 (2003)
[3] Kneip, L., Scaramuzza, D., Siegwart, R.: A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 2969-2976. IEEE, (2011)
[4] Banno, A.: A P3P problem solver representing all parameters as a linear combination. Image and Vision Computing 70, 55-62 (2018)
[5] Li, S., Xu, C., Xie, M.: A robust o (n) solution to the perspective-n-point problem. Pattern Analysis and Machine Intelligence, IEEE Transactions on 34, 1444-1450 (2012)
[6] Hesch, J., Roumeliotis, S.: A direct least-squares (DLS) method for PnP. In: IEEE International Conference on Computer Vision (ICCV), pp. 383-390. IEEE, (2011)
[7] Zheng, Y., Kuang, Y., Sugimoto, S., Astrom, K., Okutomi, M.: Revisiting the PnP Problem: A Fast, General and Optimal Solution. In: IEEE International Conference on Computer Vision (ICCV), pp. 2344-2351. (2013)
[8] Zheng, Y., Sugimoto, S., Okutomi, M.: Aspnp: An accurate and scalable solution to the perspective-n-point problem. IEICE TRANSACTIONS on Information and Systems 96, 1525-1535 (2013)

[9] Wang, P., Xu, G., Cheng, Y., Yu, Q.: A simple, robust and fast method for the perspective-n-point Problem. Pattern Recognition Letters 108, 31-37 (2018)

[10] Urban, S., Leitloff, J., Hinz, S.: MLPnP - A real-time maximum likelihood solution to the perspective-n-point problem. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 3, 131-138 (2016)

[11] Lu, C.-P., Hager, G.D., Mjolsness, E.: Fast and globally convergent pose estimation from video images. Pattern Analysis and Machine Intelligence, IEEE Transactions on 22, 610-622 (2000)

[12] Garro, V., Crosilla, F., Fusiello, A.: Solving the pnp problem with anisotropic orthogonal procrustes analysis. In: Second International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission, pp. 262-269. IEEE, (2012)

[13] Clarkson, I.V.L., Howard, S.D., Moran, W., Cochran, D., Dawson, M.L.: Maximum-likelihood and best invariant orientation estimation. In: Signals, Systems and Computers (ASILOMAR), 2010 Conference Record of the Forty Fourth Asilomar Conference on, pp. 1996-2000. (2010)

[14] Lepetit, V., Morenonoguer, F., Fua, P.: EPnP: An Accurate O(n) Solution to the PnP Problem. International Journal of Computer Vision 81, 155-166 (2009)

[15] Triggs, B., McLauchlan, P., Hartley, R., Fitzgibbon, A.: Bundle Adjustment — A Modern Synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) Vision Algorithms: Theory and Practice, vol. 1883, pp. 298-372. Springer Berlin Heidelberg (2000)

[16] Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision Second Edition. pp. 88-93. Cambridge University Press, New York (2000)