

变量投影框架下基于 Wahba 问题的多点透视问题求解算法

何颖*, 马戎, 李岁劳, 郭强

西北工业大学自动化学院, 陕西 西安 710129

摘要 提出一种在变量投影框架下的多点透视问题求解算法。多点透视问题是确定相机旋转与平移参数的基础问题, 广泛应用于计算机视觉、摄影测量、机器人、空间交会以及虚拟现实等场合。相机的旋转与平移参数在变量投影的框架下被分开估计。首先, 将平移参数固定, 根据 Wahba 问题的解法求解出最优旋转矩阵; 然后, 应用奇异值分解导数的计算公式计算当前最优旋转矩阵对平移参数的导数, 并由此计算出代价函数对平移参数的导数; 最后, 针对平移参数, 采用 Levenberg-Marquardt 算法迭代优化代价函数, 直到精度满足要求。实验结果证明, 本文算法不仅误差较小, 计算效率较高, 而且具有较大的收敛域。

关键词 机器视觉; 多点透视问题; 变量投影; Wahba 问题

中图分类号 TP391.4 文献标识码 A

doi: 10.3788/AOS201838.1115006

Solving Perspective-*n*-Point Problem in Variable Projection Framework Based on Wahba Problem

He Ying*, Ma Rong, Li Suilao, Guo Qiang

School of Automation, Northwestern Polytechnical University, Xi'an, Shaanxi 710129, China

Abstract We propose an algorithm that solves the perspective-*n*-point problem in the framework of variable projection. The perspective-*n*-point is the problem of determining the rotation and translation parameters of a camera, which is widely used in computer vision, photogrammetry, robotics, space rendezvous and virtual reality. We separately estimate the rotation and translation parameters in the variable projection framework. Firstly, an optimal rotation matrix is solved by using the Wahba problem with fixed translation parameters. By applying the formula for the derivative of the singular value decomposition, the derivative of the optimal rotation matrix is calculated with respect to the translation parameters, then the derivative of the objective function with respect to the translation parameters is obtained. Finally, the objective function is minimized by Levenberg-Marquardt algorithm over the translation parameters. The experimental results show that the proposed algorithm is one of the most accurate and effective algorithms and has a large convergence basin.

Key words machine vision; perspective-*n*-point; variable projection; Wahba problem

OCIS codes 150.1135; 150.1488; 110.4153

1 引言

多点透视问题(PnP)是根据位置已知的点与对应的像点来估计相机位姿的问题。PnP 问题是确定相机位姿的基础算法, 广泛应用于计算机视觉、摄影测量、机器人、空间交会以及虚拟现实等场合^[1]。

当点数 $n=3$ 时(P3P), 这个问题为代数方程求解问题^[2]。当 n 为一般值时, PnP 问题为优化问

题, 此时有直接求解法与迭代求解法两大类解法。

Lepetit 等^[3]提出的算法(EPnP)是一种直接求解法, 其忽略旋转矩阵的正交约束, 利用重心坐标不变性构建待估参数的方程组。Li 等^[4]提出的算法(EPnP)先选取空间中两点, 再将剩下的点分别与这两点组合, 然后按照 P3P 问题构建待估参数的方程组。Schweighofer 等^[5]提出的算法(SDP)、Hesch 等^[6]提出的算法(DLS)和 Kneip 等^[7]提出的

收稿日期: 2018-05-08; 修回日期: 2018-06-14; 录用日期: 2018-07-12

基金项目: 航空基金(20150153002)

* E-mail: cnheyng@qq.com

算法(UPnP)均使用代数几何方法求解物体空间误差的极值点,直接最小化代价函数。Zheng 等^[8]提出的算法(AsPnP)与 Zheng 等^[9]提出的算法(OPnP)也使用代数几何方法,但使用二维距离误差。直接求解法为了能够直接求解,采取简化或近似策略,往往精度较低。

迭代求解法最常见的做法是用 Levenberg-Marquardt(LM)法最小化重投影误差^[10]。Urban 等^[1]提出的算法(MLPnP)使用高斯-牛顿(GN)法最小化投影到球面切空间上的重投影误差。Lu 等^[11]的算法(LHM)与 Garro 等^[12]的算法(PPnP)使用交替最小二乘法最小化物体空间距离误差。李鑫等^[13]将 LHM 算法中重复计算的步骤提取后一次性计算,极大地提高了 LHM 算法的效率。周润等^[14]引入对位置测量噪声的加权,通过反复使用 LHM 算法调整加权权重,从而优化重投影误差。迭代算法一般精度较高。然而,迭代法需要初始值,如果初始值没有落在收敛域中,算法将不收敛,这扩大收敛域将变得有意义。

最近,Hong 等^[15]将变量投影法应用于光束调整法,扩大了光束调整法的收敛域^[16]。遗憾的是,该方法还不能够处理已标定相机。实际上,对于已标定相机,LHM 与 PPnP 这两个更早期的算法已经使用了变量投影的思想。但是,LHM 与 PPnP 根据绝对定向问题计算旋转参数,误差较大。最小化物体空间误差的绝对定向问题与最小化球面距离误差的 Wahba 问题^[17]关系密切。球面距离误差比物体空间误差更接近理想的重投影误差。基于此,本文在变量投影框架下,先根据 Wahba 问题直接求解最优的旋转参数,再应用 LM 算法优化剩下的平移参数。这样既可以扩大收敛域,又可以使用更理想的误差模型,而且初始平移参数比初始旋转参数更容易测量。

2 问题描述及解法

2.1 PnP 问题

成像过程可以分解为坐标变换与投影成像两个步骤。采用的坐标变换公式为

$$\mathbf{q}_i = \mathbf{R}(\mathbf{p}_i + \mathbf{t}), \quad (1)$$

投影成像公式为

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} q_x/q_z \\ q_y/q_z \end{pmatrix} = \begin{bmatrix} \mathbf{r}_1^T(\mathbf{p}_i + \mathbf{t})/\mathbf{r}_3^T(\mathbf{p}_i + \mathbf{t}) \\ \mathbf{r}_2^T(\mathbf{p}_i + \mathbf{t})/\mathbf{r}_3^T(\mathbf{p}_i + \mathbf{t}) \end{bmatrix}, \quad (2)$$

式中: \mathbf{p}_i 是参考点 i 在世界坐标系下的坐标($i=1, \dots, n$); $\mathbf{q}_i=(q_x, q_y, q_z)^T$ 是参考点 i 在相机坐标系下的坐标; u_i 与 v_i 分别对应参考点 i 的水平与垂直方向的归一化图像坐标; $\mathbf{R}=(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3)^T$ 是世界坐标系到相机坐标系的旋转矩阵; $\mathbf{t}=(t_x, t_y, t_z)^T$ 是世界坐标系到相机坐标系的平移矢量。通过 u_i 与 v_i 可以计算出从光心指向像点的方向矢量 $\mathbf{v}_i=(u_i, v_i, 1)^T/\sqrt{u_i^2+v_i^2}$ 。PnP 问题就是根据 \mathbf{p}_i 与 \mathbf{v}_i 来计算 \mathbf{R} 与 \mathbf{t} 的问题。

实际系统往往有测量噪声。假定单位方向矢量的测量值 $\mathbf{v}_i \sim N[\bar{\mathbf{v}}_i, \sigma^2(\mathbf{I}_3 - \bar{\mathbf{v}}_i \bar{\mathbf{v}}_i^T)]$, 式中 $\bar{\mathbf{v}}_i$ 为方向矢量的真值, \mathbf{I} 为单位矩阵。求解 \mathbf{R} 与 \mathbf{t} 的问题归结为最小化球面误差的最小二乘问题:

$$\{\mathbf{R}^*, \mathbf{t}^*\} = \arg \min_{\mathbf{R} \in SO(3), \mathbf{t}} J(\mathbf{R}, \mathbf{t}) \triangleq \arg \min_{\mathbf{R} \in SO(3), \mathbf{t}} \sum_{i=1}^n \| \mathbf{v}_i - \mathbf{R}(\mathbf{p}_i + \mathbf{t}) / \| \mathbf{R}(\mathbf{p}_i + \mathbf{t}) \| \|^2, \quad (3)$$

式中: $\|\cdot\|$ 为 2-范数; $\det(\cdot)$ 为行列式; $SO(3)$ 为特殊正交群。

2.2 变量投影

变量投影是一种将非线性最小二乘问题的待估参数分组优化的算法框架。变量投影法将待估参数分成两组,根据最优条件先解出两组参数间的关系,从原代价函数中消去一组参数形成新的代价函数,最后优化新的代价函数,得到原问题的最优解。变量投影减少了待估参数的数量,不仅使得算法更高效,而且有很大概率使算法的收敛域^[15,18]。

变量投影最典型的应用是解决如下非线性最小二乘问题^[19]:

$$\min_{\mathbf{u}, \mathbf{v}} \| \mathbf{G}(\mathbf{u})\mathbf{v} - \mathbf{z}(\mathbf{u}) \|^2, \quad (4)$$

式中 $\mathbf{u} \in \mathbb{R}^p$, $\mathbf{v} \in \mathbb{R}^q$, $\mathbf{G}: \mathbb{R}^p \rightarrow \mathbb{R}^{s \times q}$, $\mathbf{z}: \mathbb{R}^p \rightarrow \mathbb{R}^s$ 。

变量投影将(4)式对 \mathbf{u}, \mathbf{v} 的优化转换为只对 \mathbf{u} 的优化。首先,注意到给定 \mathbf{u} 时 \mathbf{v} 的最优值为

$$\mathbf{v}^*(\mathbf{u}) = \arg \min_{\mathbf{v}} \| \mathbf{G}(\mathbf{u})\mathbf{v} - \mathbf{z}(\mathbf{u}) \| = \mathbf{G}(\mathbf{u})^\dagger \mathbf{z}(\mathbf{u}), \quad (5)$$

式中 \mathbf{G}^\dagger 表示 \mathbf{G} 的伪逆。代入(4)式得到简化的问题:

$$\begin{aligned} \min_{\mathbf{u}} \| \boldsymbol{\varepsilon}^*(\mathbf{u}) \|^2 &= \min_{\mathbf{u}} \| \boldsymbol{\varepsilon}^*[\mathbf{u}, \mathbf{v}^*(\mathbf{u})] \|^2 = \\ &\min_{\mathbf{u}} \| [\mathbf{G}(\mathbf{u})\mathbf{G}(\mathbf{u})^\dagger - \mathbf{I}] \mathbf{z}(\mathbf{u}) \|^2. \end{aligned} \quad (6)$$

(6)式也可以视为一个子空间优化问题:

$$\begin{aligned} \boldsymbol{\varepsilon}^*(\mathbf{u}) &= [\mathbf{I} - \mathbf{G}(\mathbf{u})\mathbf{G}(\mathbf{u})^\dagger][\mathbf{G}(\mathbf{u})\mathbf{v} - \mathbf{z}(\mathbf{u})] = \\ &[\mathbf{I} - \mathbf{J}_v(\mathbf{u})\mathbf{J}_v(\mathbf{u})^\dagger]\boldsymbol{\varepsilon}(\mathbf{u}, \mathbf{v}). \end{aligned} \quad (7)$$

变量投影采用 LM 优化算法,需要计算简化问题的残差,即(7)式对 \mathbf{u} 的导数:

$$\frac{d\boldsymbol{\varepsilon}^*(\boldsymbol{u})}{d\boldsymbol{u}} = \frac{\partial \boldsymbol{\varepsilon}^*[\boldsymbol{u}, \boldsymbol{v}^*(\boldsymbol{u})]}{\partial \boldsymbol{v}} \frac{d\boldsymbol{v}^*(\boldsymbol{u})}{d\boldsymbol{u}} + \frac{\partial \boldsymbol{\varepsilon}^*[\boldsymbol{u}, \boldsymbol{v}^*(\boldsymbol{u})]}{\partial \boldsymbol{u}}, \quad (8)$$

式中 $\frac{\partial \boldsymbol{\varepsilon}^*[\boldsymbol{u}, \boldsymbol{v}^*(\boldsymbol{u})]}{\partial \boldsymbol{v}}$ 与 $\frac{\partial \boldsymbol{\varepsilon}^*[\boldsymbol{u}, \boldsymbol{v}^*(\boldsymbol{u})]}{\partial \boldsymbol{u}}$ 是原问题的导数, $\frac{d\boldsymbol{v}^*(\boldsymbol{u})}{d\boldsymbol{u}} \approx -\boldsymbol{J}_v(\boldsymbol{u})^\dagger \boldsymbol{J}_u[\boldsymbol{u}, \boldsymbol{v}^*(\boldsymbol{u})]$ 。

注意,前面的推导虽然针对的是非线性最小二乘问题,但对于矩阵分解^[20]、光束平差^[16]等问题也适用。后面给出将变量投影应用到 PnP 问题的具体方法。

2.3 最优旋转矩阵

Wahba 问题是根据两个观测坐标系下方向矢量的对应观测值确定两个坐标系间最优旋转矩阵的问题。用 Wahba 问题求解(3)式中的 R 。

注意到旋转矩阵不改变长度,即 $\|\boldsymbol{R}(\boldsymbol{p}_i + \boldsymbol{t})\| = \|\boldsymbol{p}_i + \boldsymbol{t}\|$, (3)式可以分解为内层对 \boldsymbol{R} 的优化:

$$\{\boldsymbol{R}^*(\boldsymbol{t})\} = \arg \min_{\boldsymbol{R} \in SO(3)} \sum_{i=1}^n \|\boldsymbol{v}_i - \boldsymbol{R}(\boldsymbol{p}_i + \boldsymbol{t})/\|\boldsymbol{p}_i + \boldsymbol{t}\|\|^2, \quad (9)$$

与外层对 \boldsymbol{t} 的优化:

$$\{\boldsymbol{t}^*\} = \arg \min_{\boldsymbol{t}} J[\boldsymbol{R}^*(\boldsymbol{t}), \boldsymbol{t}] \triangleq \arg \min_{\boldsymbol{t}} \sum_{i=1}^n \|\boldsymbol{v}_i - \boldsymbol{R}^*(\boldsymbol{t})(\boldsymbol{p}_i + \boldsymbol{t})/\|\boldsymbol{p}_i + \boldsymbol{t}\|\|^2. \quad (10)$$

$$\begin{cases} \frac{d\boldsymbol{r}_i}{dt} = -\boldsymbol{R}^* \boldsymbol{D}_i - \frac{d\boldsymbol{R}^*}{d\boldsymbol{A}} \left[\sum_{j=1}^n (\boldsymbol{v}_j \otimes \boldsymbol{I}) \boldsymbol{D}_j \right] \frac{(\boldsymbol{p}_i + \boldsymbol{t})}{\|\boldsymbol{p}_i + \boldsymbol{t}\|}, \\ \boldsymbol{D}_i = [\boldsymbol{I}_3 - (\boldsymbol{p}_i + \boldsymbol{t})(\boldsymbol{p}_i + \boldsymbol{t})^\top / (\boldsymbol{p}_i + \boldsymbol{t})^\top (\boldsymbol{p}_i + \boldsymbol{t})] / \|\boldsymbol{p}_i + \boldsymbol{t}\| \end{cases}, \quad (13)$$

式中 \otimes 是克罗内克积, \boldsymbol{A} 与 \boldsymbol{R}^* 均由(11)式得出。

注意到(11)式中的 $\det(\boldsymbol{U}\boldsymbol{V}^\top)$ 是一个用于符号校正的常值系数,可以得到:

$$\frac{d\boldsymbol{R}^*}{d\boldsymbol{A}} = \boldsymbol{U} \frac{d\boldsymbol{V}^\top}{d\boldsymbol{A}} + \frac{d\boldsymbol{U}}{d\boldsymbol{A}} \boldsymbol{V}^\top, \quad (14)$$

式中 \boldsymbol{U} 与 \boldsymbol{V} 是对作 \boldsymbol{A} 奇异值分解得到的两个正交矩阵,均由(11)式得出。Papadopoulos 等^[22]给出了 $\frac{d\boldsymbol{V}^\top}{d\boldsymbol{A}}$ 与 $\frac{d\boldsymbol{U}}{d\boldsymbol{A}}$ 的分量形式:

$$\begin{cases} \frac{\partial \boldsymbol{U}}{\partial \boldsymbol{A}_{mn}} = \boldsymbol{U} \boldsymbol{\Omega}_U^{mn}, \\ \frac{\partial \boldsymbol{V}^\top}{\partial \boldsymbol{A}_{mn}} = -\boldsymbol{\Omega}_V^{mn\top} \boldsymbol{V} \end{cases}, \quad (15)$$

式中 $\boldsymbol{\Omega}_U^{mn}$ 与 $\boldsymbol{\Omega}_V^{mn}$ 是线性方程的解:

$$\begin{cases} s_m \boldsymbol{\Omega}_{Ukl}^{mn} + s_k \boldsymbol{\Omega}_{Vkl}^{mn} = \boldsymbol{U}_{mk} \boldsymbol{V}_{nl}, \\ s_k \boldsymbol{\Omega}_{Ukl}^{mn} + s_l \boldsymbol{\Omega}_{Vkl}^{mn} = -\boldsymbol{U}_{ml} \boldsymbol{V}_{nk} \end{cases}, \quad (16)$$

(9)式为标准的 Wahba 问题,Markley^[17] 给出了旋转矩阵的最优解:

$$\begin{cases} \boldsymbol{R}^* = \boldsymbol{U} \operatorname{diag}[1 \ 1 \ \det(\boldsymbol{U}\boldsymbol{V}^\top)] \boldsymbol{V}^\top \\ \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top = \operatorname{SVD}(\boldsymbol{A}) \\ \boldsymbol{A} \triangleq \sum_{i=1}^n \boldsymbol{v}_i(\boldsymbol{p}_i + \boldsymbol{t})^\top / \|\boldsymbol{p}_i + \boldsymbol{t}\| \end{cases}, \quad (11)$$

式中 SVD(\cdot) 为奇异值分解, $\operatorname{diag}[\cdot]$ 表示对角矩阵。当需要强调 \boldsymbol{R}^* 是 t 的函数时,也将 \boldsymbol{R}^* 记为 $\boldsymbol{R}^*(t)$ 。

2.4 最优平移参数

前面,根据 Wahba 问题已经求解出最优旋转矩阵 $\boldsymbol{R}^*(t)$, 将原问题变为估计最优平移参数 t^* 的问题:

$$\begin{cases} \{\boldsymbol{t}^*\} = \arg \min_t \sum_{i=1}^n \|\boldsymbol{r}_i\|_2^2 \\ \boldsymbol{r}_i = \boldsymbol{v}_i - \boldsymbol{R}^*(t)(\boldsymbol{p}_i + \boldsymbol{t}) / \|\boldsymbol{p}_i + \boldsymbol{t}\| \end{cases}, \quad (12)$$

式中 $\boldsymbol{R}^*(t)$ 由(11)式得出。

LM 算法的关键在于计算 $\frac{d\boldsymbol{r}_i}{dt}$, 这需要计算矩阵

对矩阵函数的导数。选择使用 Magnus 等^[21] 的定义: $\frac{\partial \boldsymbol{Y}}{\partial \boldsymbol{X}} = \frac{\partial \operatorname{vec}(\boldsymbol{Y})}{\partial \operatorname{vec}(\boldsymbol{X})}$, 式中 $\operatorname{vec}(\cdot)$ 表示矩阵按列拉直。由链式法则得到:

式中: s_1, s_2, s_3 为降序排列的矩阵 \boldsymbol{A} 的特征值; $m, n, k, l = 1, 2, 3$ 。

将(14)式代入(13)式完成 $\frac{d\boldsymbol{r}_i}{dt}$ 计算。应用 LM 算法计算出最优平移矢量 \boldsymbol{t}^* , 即可得最优旋转矩阵 $\boldsymbol{R}^*(t^*)$ 。

3 实验结果与分析

将本文算法从精度、收敛域与计算时间等角度与典型算法进行比较。用于对比的算法及选取理由如下: 1) LM, 用 LM 算法最小化重投影误差, 是常规但能够达到理论精度的算法^[10]; 2) RPnP, 是通过将点分组再按照 P3P 问题构建代数方程组的算法, 代价函数是代数方程的残差平方和; 3) EPnP+GN, 采用代数方法最小化代数误差, 然后使用 GN 法最

小化虚拟控制点的几何误差,运算效率较高^[3];4) DLS++,采用代数几何方法最小化物体空间误差,三个加号表示包含了 DLS 算法的作者使用 3 次预旋转以避免原算法在某些退化角度情况下无法正确求解的修正算法;5) OPnP,采用代数几何方法最小化二维物体空间误差,是发表时间较晚、精度较高的代数几何类算法^[9];6) LHM^[11]与 PPnP^[12],使用物体空间距离误差的算法,计算旋转矩阵的方式与本文算法非常类似;7) MLPnP,采用高斯-牛顿迭代法优化投影到球面的重投影误差,是最近发表的具有实时计算能力的迭代类算法^[1]。

除了 LM 采用 Zheng 等^[9]在 OPnP 工具箱中的算法实现外,其他算法全部采用原论文作者公开的算法实现。除了数据输入输出格式上的适配之外,原则上不对代码做实质改动。另外,为了只比较元算法,消除初始化的影响,在比较计算误差与运行时间时,迭代算法的初始值统一用直接线性变换法^[10]计算。

仿真的相机焦距为 1000 pixel,图像为 800 pixel × 600 pixel,主点位于图像中心。参考点的 xyz 坐标在相机坐标系中 [−1,1] × [−1,1] × [1,4] m 的

范围按照均匀分布随机产生,相机位置的 xyz 坐标在世界坐标系中 [−10,10] × [−10,10] × [−10,10] m 的范围内按照均匀分布随机产生,相机姿态的三个 Rodriguez 角均在 [−π, π] 范围内按均匀分布随机产生。图像噪声的均方差设置为 5 pixel。

3.1 计算误差

图 1 与图 2 分别为参考点数量从 10 变到 100 时的旋转误差与平移误差。平移误差与角度误差分别为 $\| t - t_{\text{true}} \|$ 与 $\| \text{Euler}(\mathbf{R}/\mathbf{R}_{\text{true}}) \|$, 式中 Euler(·) 为计算旋转矩阵的欧拉角。图中所示为 10000 次独立仿真的平均结果。

由图 1 与图 2 可以看出,无论是旋转误差还是平移误差,整体上均随着参考点数量的增大而减小。RPnP 与 EPnP+GN 算法的误差最大,这可能是由于这两种直接计算方法使用的代价函数过于简化。DLS++、PPnP、OPnP 与 LHM 算法的误差非常接近,这是因为它们均使用物体空间误差。误差最小的是 LM、MLPnP 算法与本文算法。LM 算法使用重投影误差,MLPnP 算法使用投影到球面的重投影误差,本文算法使用球面误差,其误差曲线几乎完全重叠,说明球面误差与重投影误差非常接

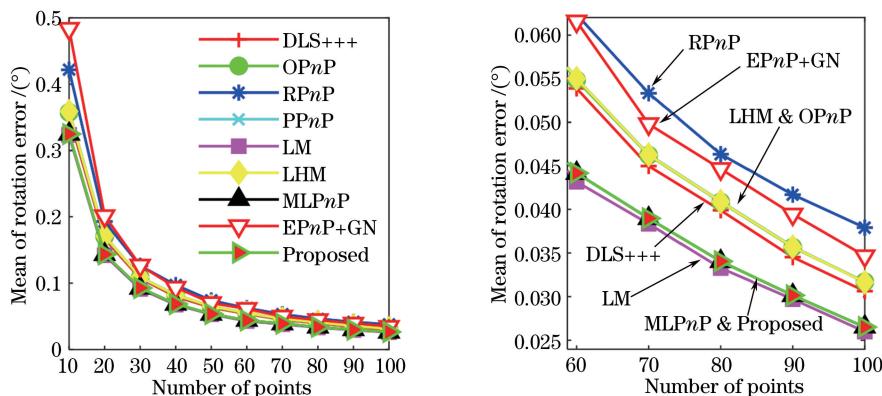


图 1 不同参考点数目时的(a)平均旋转误差和(b)局部放大

Fig. 1 (a) Mean of rotation error and (b) regional magnified view versus number of points

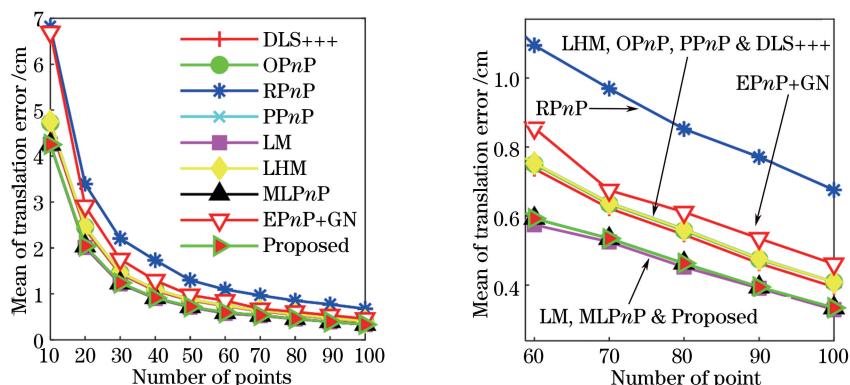


图 2 不同参考点数目时的(a)平均平移误差和(b)局部放大

Fig. 2 (a) Mean of translation error and (b) regional magnified view versus number of points

近。由图 1 与图 2 可以看出,无论参考点数量为 10 还是 100,DLS+++,PPnP、OPnP 与 LHM 算法的误差均比本文算法的误差高约 20%,这一现象与 Lu 等^[11]认为物体空间误差是“有偏的”结论一致。

3.2 收敛域

算法的收敛域一般无法直接测量,因此,通过增加初始值的噪声,比较噪声激励出误差大小的方式比较收敛域的大小。噪声大小相同时,收敛域大的算法的初始值不易落在收敛域外,收敛到正确解的概率更大,平均误差较小;反之,收敛域小的算法平均误差较大。

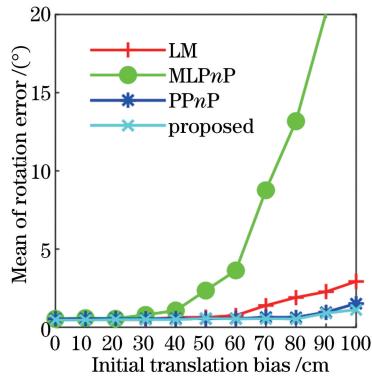


图 3、图 4 分别为平移参数初始值的噪声标准差由 0.2 m 增大到 2 m 时各迭代算法的平均旋转误差与平均平移误差。注意,RPnP、OPnP 与 DLS+++ 算法不存在收敛域,而 LHM 算法不需要平移初始值,因此这些算法并未纳入其中。从图 3 与图 4 中可以看出,参考点数量较多时,各个算法均相对不易受初始化噪声影响。当参考点数量相同时,使用全部参数迭代的 LM 与 MLPnP 比使用变量投影的 PPnP 及本文算法更易受到初始化噪声的干扰。这与 Hong 等^[16]、Okatani 等^[20]和 O’Leary 等^[19]在其他应用场景中观察到的变量投影法可以扩大收敛域的结论一致。

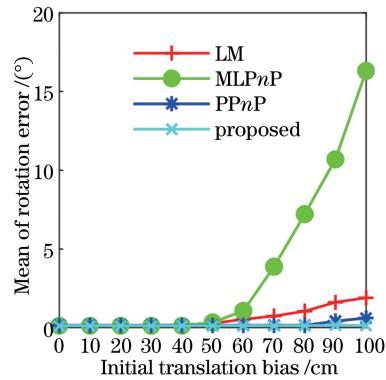


图 3 不同初始平移误差时的平均旋转误差。(a) 10 个参考点;(b) 100 个参考点

Fig. 3 Mean of rotation error versus initial translation bias. (a) 10 points; (b) 100 points

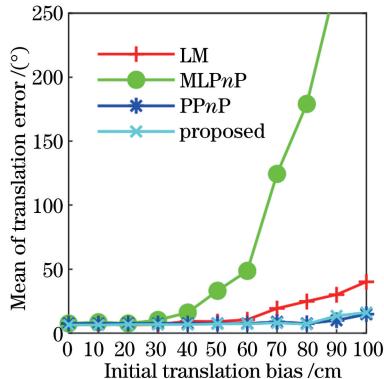


图 4 不同初始平移误差时的平均平移误差。(a) 10 个参考点;(b) 100 个参考点

Fig. 4 Mean of translation error versus initial translation bias. (a) 10 points; (b) 100 points

3.3 运算效率

采用算法的执行时间来度量运算效率。图 5 所示为参考点个数从 10 增加到 100 时各算法的执行时间。运行环境为:Windows 7,Intel i7 4720HQ 2.6 G,8 GB RAM,Matlab R2018a。从图 5 中可以看出,执行时间较长的是 OPnP 与 DLS+++ 这两种代数几何算法,执行时间较短的是 EPnP+GN 与 RPnP 这两种直接求解算法,因为它们的误差准则非常简单。本文算法是执行时间最短的迭代类算

法,所用时间与 EPnP+GN 非常接近。LHM 和 PPnP 虽然与本文算法在求解旋转矩阵的方法上类似,但是较本文算法慢。这可能是由于它们在优化平移参数时采用交替最小二乘法,完全忽略了两组参数间的联系,导致收敛性下降,执行时间变长。LM 与 MLPnP 比本文算法慢可能是由于这两个算法采用的联合参数优化方法的参数空间维度高,雅可比矩阵的阶数也高,运算效率因此降低。

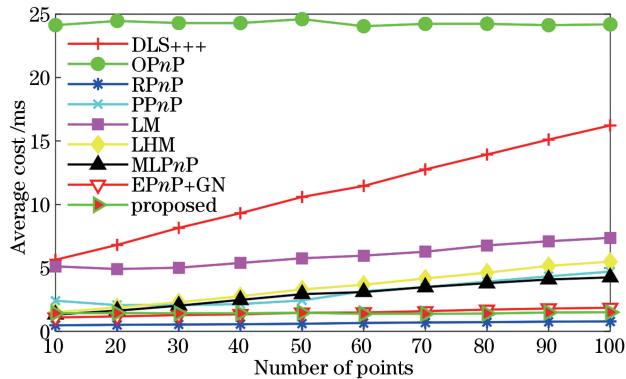


图 5 不同参考点数目时的运行时间

Fig. 5 Execution time versus number of points

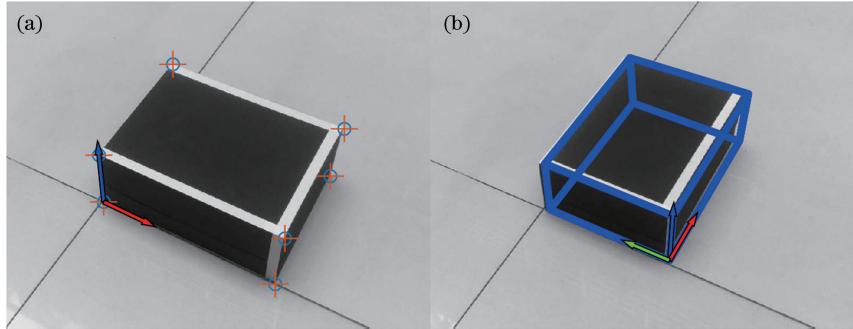


图 6 真实图像检验算法正确性。(a)原始位置;(b)平移旋转后

Fig. 6 Verification of proposed algorithm with real images. (a) Original position; (b) after translation and rotation

4 结 论

多点透视问题是确定相机位姿的基础算法。本文提出的多点透视投影问题求解算法有三个特点：1)采用球面误差，结果显示旋转和平移误差均较小，与理想的重投影误差几乎没有差别；2)通过与 Wahba 问题相结合，获得了较高的计算效率；3)采用变量投影框架，只需要初始化平移参数，结果显示收敛域较大，不易受初始化误差影响。虽然通常情况下平移参数比旋转参数更容易测量，但是只能用平移参数初始化、不能用旋转参数初始化也是本文算法的不足之处。

参 考 文 献

- [1] Urban S, Leitloff J, Hinz S. MLPnP: a real-time maximum likelihood solution to the perspective- n -point problem [J]. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2016, III-3: 131-138.
- [2] Wang P, Xu G L, Wang Z S, et al. An efficient solution to the perspective-three-point pose problem[J]. Computer Vision and Image Understanding, 2018, 166: 81-87.
- [3] Lepetit V, Moreno-Noguer F, Fua P. EPnP: an accurate $O(n)$ solution to the PnP problem [J]. International Journal of Computer Vision, 2008, 81(2): 155-166.
- [4] Li S Q, Xu C, Xie M. A robust $O(n)$ solution to the perspective- n -point problem [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(7): 1444-1450.
- [5] Schweighofer G, Pinz A. Globally optimal $O(n)$ solution to the PnP problem for general camera models [C]// British Machine Vision Conference, 2008: 1-10.
- [6] Hesch J A, Roumeliotis S I. A direct least-squares (DLS) method for PnP [C]// IEEE International Conference on Computer Vision, 2011: 383-390.
- [7] Kneip L, Li H D, Seo Y. UPnP: an optimal $O(n)$ solution to the absolute pose problem with universal applicability[C]// European Conference on Computer Vision, 2014: 127-142.
- [8] Zheng Y Q, Sugimoto S, Okutomi M. ASPnP: an accurate and scalable solution to the perspective- n -point problem[J]. IEICE Transactions on Information and Systems, 2013, 96(7): 1525-1535.
- [9] Zheng Y Q, Kuang Y B, Sugimoto S, et al. Revisiting the pnp problem: a fast, general and optimal solution[C]// IEEE International Conference on Computer Vision, 2013: 2344-2351.

- [10] Hartley R, Zisserman A. Multiple view geometry in computer vision[M]. 2nd ed. Cambridge: Cambridge University Press, 2003: 88-93.
- [11] Lu C P, Hager G D, Mjolsness E. Fast and globally convergent pose estimation from video images[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(6): 610-622.
- [12] Garro V, Crosilla F, Fusiello A. Solving the pnp problem with anisotropic orthogonal procrustes analysis[C]// Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 2012: 262-269.
- [13] Li X, Long G C, Liu J B, et al. Accelerative orthogonal iteration algorithm for camera pose estimation[J]. Acta Optica Sinica, 2015, 35(1): 0115004.
李鑫, 龙古灿, 刘进博, 等. 相机位姿估计的加速正交迭代算法[J]. 光学学报, 2015, 35(1): 0115004.
- [14] Zhou R, Zhang Z Y, Huang X H. Weighted orthogonal iteration algorithm for camera pose estimation [J]. Acta Optica Sinica, 2018, 38(5): 0515002.
周润, 张征宇, 黄叙辉. 相机位姿估计的加权正交迭代算法[J]. 光学学报, 2018, 38(5): 0515002.
- [15] Hong J H, Zach C, Fitzgibbon A. Revisiting the variable projection method for separable nonlinear least squares problems [C]// IEEE Conference on Computer Vision and Pattern Recognition, 2017: 5939-5947.
- [16] Hong J H, Zach C, Fitzgibbon A, et al. Projective bundle adjustment from arbitrary initialization using the variable projection method [C]// European Conference on Computer Vision, 2016: 477-493.
- [17] Markley F L. Attitude determination using vector observations and the singular value decomposition [J]. Journal of the Astronautical Sciences, 1988, 38(3): 245-258.
- [18] Strelow D. General and nested Wiberg minimization: L2 and maximum likelihood [C]// European Conference on Computer Vision, 2012: 195-207.
- [19] O'Leary D P, Rust B W. Variable projection for nonlinear least squares problems [J]. Computational Optimization and Applications, 2013, 54 (3): 579-593.
- [20] Okatani T, Yoshida T, Deguchi K. Efficient algorithm for low-rank matrix factorization with missing components and performance comparison of latest algorithms [C]// International Conference on Computer Vision, 2011: 842-849.
- [21] Magnus J R, Neudecker H. Matrix differential calculus with applications in statistics and econometrics[M]. West Sussex: Wiley, 1988: 214-215.
- [22] Papadopoulou T, Lourakis M I A. Estimating the jacobian of the singular value decomposition: Theory and applications [C]// European Conference on Computer Vision, 2000: 554-570.