

〈系统与设计〉

基于扩展正交迭代的快速加权的相机位姿估计

王世安, 王向军, 阴 雷

(天津大学 精密测试技术及仪器国家重点实验室, 天津 300072)

摘要: 相机位姿估计算法多基于参考点而较少利用图像中的直线信息, 本文对于相机位姿估计算法的抗干扰性和实时性, 在扩展正交迭代的基础上, 提出了一种基于点和直线段结合的快速加权的相机位姿估计算法, 该算法以加权共线性误差和加权共面性误差之和为误差函数, 根据计算初值的深度信息和重投影误差确定权重系数, 并对整体进行加速优化, 将每次迭代计算的时间复杂度从 $O(n)$ 降到了 $O(1)$ 。仿真实验结果表明算法可以抑制异常点的干扰, 减少计算时间, 旋转矩阵计算误差比传统正交迭代算法减少 48.31%, 平移向量计算误差减少 48.79%, 加速优化后的计算时间为加速前的 47.11%。实物实验表明该算法可以充分利用检测到的参考点和参考直线信息, 提高计算精度, 有较高的实际应用价值。

关键词: 机器视觉; 图像处理; 位姿估计; 迭代算法; 计算复杂度

中图分类号: TP391

文献标识码: A

文章编号: 1001-8891(2020)05-0205-08

Accelerative and Weighted Camera Pose Estimation Based on Extended Orthogonal Iterative Algorithm

WANG Shi'an, WANG Xiangjun, YIN Lei

(State Key Laboratory of Precision Measurement Technology and Instruments, Tianjin University, Tianjin 300072, China)

Abstract: The camera pose estimation algorithm is typically more based on reference points and less on the linear information in the image. In this paper, based on the extended orthogonal iteration, an accelerative and weighted camera pose estimation algorithm based on the combination of point and line segments is proposed to ensure a good anti-interference and real-time performance of the algorithm. The algorithm considers the sum of the weighted collinearity error and the weighted coplanarity error as the error function. The weight coefficients are determined by the depth information and the re-projection error of the initial value. In addition, the algorithm accelerates the iterative process and reduces the time complexity of the iteration process dramatically from $O(n)$ to $O(1)$. The simulation results demonstrate that the algorithm can suppress the interference of abnormal points and reduce the calculation time. The calculation accuracy of the rotation matrix is 51.69% higher than that of the traditional orthogonal iterative algorithm, and the calculation accuracy of the translation vector is increased by 51.21%. Furthermore, the calculation time after the acceleration is reduced by 47.11%. The physical experiment shows that the algorithm can make full use of the detected reference points and reference line information, improve the calculation accuracy, reduce the cumulative error, and has high practical application value.

Keywords: machine vision, image processing, pose estimation, iterative algorithms, computational complexity

0 引言

相机位姿估计问题是摄影测量学^[1]、计算机视觉^[2]、

计算机图形学^[3]以及机器人科学^[4]的一个重要问题, 在无人平台导航^[5]、航天器对接^[6]、相机标定^[7]、增强现实^[8]等领域有广泛应用。位姿估计问题根据相对应

收稿日期: 2019-10-09; 修订日期: 2020-02-27.

作者简介: 王世安(1995-), 男, 硕士, 主要研究方向为计算机视觉和图像处理。E-mail: sean_wang@tju.edu.cn。

特征信息的不同可以分为3类^[9]：2D-2D位姿估计、2D-3D位姿估计、3D-3D位姿估计。其中2D-3D位姿估计问题又叫做 n 点透视问题（PnP, Perspective-n-Point）。

目前在PnP问题的迭代算法中，文献[10]提出的正交迭代算法计算精度高，迭代次数少，计算速度快并且有全局收敛性。直线作为特征点之外易于获取且较为稳定的几何特征在位姿估计中可以提高算法精度和鲁棒性，并且在缺少纹理、图案重复的情况下可以起到主导作用^[11]。此外在复杂真实环境中进行实时位姿估计时，由于测量误差难免会产生异常数据，正交迭代算法的计算结果会有较大偏差。本文对于文献[12]提出的基于点和直线段对应的扩展正交迭代算法进行了优化，通过初值得到普通特征点和直线段上特征点的深度和重投影误差，通过初值的深度信息和重投影误差确定物方空间误差的权重系数，并根据文献[13]提出的思想，提出了在点和直线段对应情况下的快速加权正交迭代算法，将算法迭代时的计算复杂度从 $O(n)$ 降低到 $O(1)$ ，在提高计算速度的同时，提高了算法的抗干扰能力和旋转矩阵、平移向量的计算精度，并可以克服场景中缺少特征点的情况。

1 基于点和直线段结合的快速加权正交迭代算法

1.1 扩展正交迭代算法

正交迭代算法^[10]计算参考点 p_i 的共线性误差，并以此作为迭代计算的误差函数，即：

$$E(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^n \|e_i\|^2 = \sum_{i=1}^n \|(\mathbf{I} - \mathbf{V}_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})\|^2 \quad (1)$$

令 $\mathbf{v}_i$ 为参考点 p_i 在归一化平面上的投影，则公式中 $\mathbf{I}$ 为单位矩阵； $\mathbf{V}_i = \mathbf{v}_i \mathbf{v}_i^T / (\mathbf{v}_i^T \mathbf{v}_i)$ 为视线投影矩阵； $\mathbf{R}$ 为旋转矩阵； $\mathbf{t}$ 为平移向量，且旋转矩阵 $\mathbf{R}$ 为正交矩阵，有 $\mathbf{R}\mathbf{R}^T = \mathbf{I}$ 。

扩展正交迭代算法^[12]除了计算参考点在物体空间的共线性误差以外，还计算参考直线段在物体空间的共面性误差。如图1所示， l_i 为参考直线段 L_i 在归一化图像平面上的投影，则在理想情况下直线 L_i 、直线 l_i 和 O' 共面于平面 N_i 。如图2所示，设在平面 N_i 上有向量 $\mathbf{X}^{L_i}$ 和向量 $\mathbf{Y}^{L_i}$ 相互垂直，且 $\mathbf{X}^{L_i}$ 和 $\mathbf{Y}^{L_i}$ 的单位向量分别为 $\mathbf{h}_i$ 和 $\mathbf{g}_i$ ，则参考直线端点 p_i 在 $\mathbf{X}^{L_i}$ 方向的投影为 $\mathbf{H}_i(\mathbf{R}\mathbf{p}_i + \mathbf{t})$ ，其中 $\mathbf{H}_i = \mathbf{h}_i \mathbf{h}_i^T / (\mathbf{h}_i^T \mathbf{h}_i)$ 为 $\mathbf{X}^{L_i}$ 方向的视线投影矩阵，参考点在 $\mathbf{Y}^{L_i}$ 方向的投影为 $\mathbf{G}_i(\mathbf{R}\mathbf{p}_i + \mathbf{t})$ ，其中 $\mathbf{G}_i = \mathbf{g}_i \mathbf{g}_i^T / (\mathbf{g}_i^T \mathbf{g}_i)$ 为 $\mathbf{Y}^{L_i}$ 方向的视线

投影矩阵。

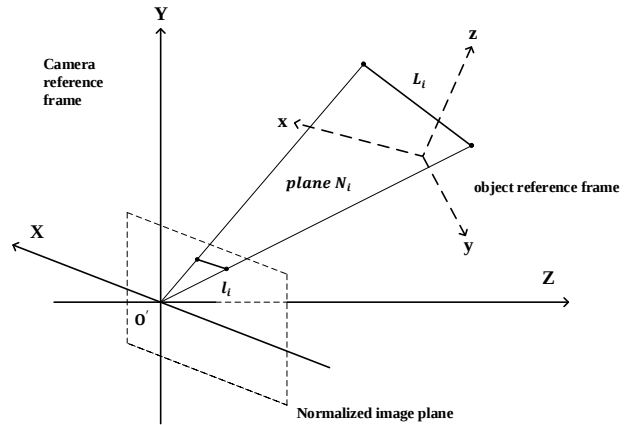


图1 物体空间和归一化平面上的对应直线段

Fig.1 Object space and corresponding line segments on the normalized plane

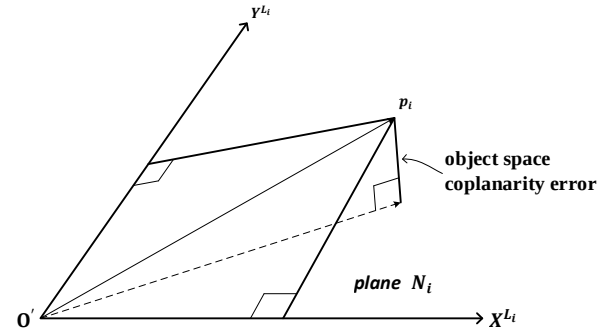


图2 物体空间的共面性误差

Fig.2 Coplanarity error of object space

如图2所示，参考直线端点在平面 N_i 的投影为 $\mathbf{U}_i(\mathbf{R}\mathbf{p}_i + \mathbf{t}) = (\mathbf{H}_i + \mathbf{G}_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})$ ，由此得到参考点 p_i 与平面 N_i 的共面性误差为：

$$e_i = (\mathbf{I} - \mathbf{U}_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t}) \quad (2)$$

令直线段 L_i 的两个端点为 $p_1^{L_i}$ 和 $p_2^{L_i}$ ，则 $p_1^{L_i}$ 与平面 N_i 的共面性误差为：

$$e_1^{L_i} = (\mathbf{I} - \mathbf{U}_i)(\mathbf{R}\mathbf{p}_1^{L_i} + \mathbf{t}) \quad (3)$$

$p_2^{L_i}$ 与平面 N_i 的共面性误差为：

$$e_2^{L_i} = (\mathbf{I} - \mathbf{U}_i)(\mathbf{R}\mathbf{p}_2^{L_i} + \mathbf{t}) \quad (4)$$

则将直线段 L_i 在物体空间的误差的平方定义为其两个端点的共面性误差的平方之和：

$$\|e_{L_i}\|^2 = \|e_1^{L_i}\|^2 + \|e_2^{L_i}\|^2 = \sum_{j=1}^n \|(\mathbf{I} - \mathbf{U}_i)(\mathbf{R}\mathbf{p}_j^{L_i} + \mathbf{t})\|^2 \quad (5)$$

设位姿估计前可以提取到 n 个参考点($p_1, p_2, \dots, p_n$)和 m 个参考直线段($L_1, L_2, \dots, L_m$)，则扩展正交迭代算法的目标函数为：

$$E(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^n \|e_i\|^2 + \sum_{i=1}^m \|e_{l_i}\|^2 = \sum_{i=1}^n \|(I - V_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})\|^2 + \sum_{i=1}^m \sum_{j=1}^2 \|(I - U_i)(\mathbf{R}\mathbf{p}_j^{l_i} + \mathbf{t})\|^2 \quad (6)$$

$$= \sum_{i=1}^N \|(I - W_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})\|^2$$

$$W_i = \begin{cases} V_i, & i = j, j = 1, 2, \dots, n \\ U_i, & i = \text{INT}\left(\frac{j-n+1}{2}\right), j = n+1, n+2, \dots, n+2*m \end{cases} \quad (7)$$

式中: $N=n+2*m$, $\mathbf{R}\mathbf{R}^T=\mathbf{I}$, INT 表示取整数。

1.2 加权扩展正交迭代

1.2.1 加权扩展正交迭代算法

扩展正交迭代对于每一个参考点和每一条检测到的直线段都赋予相同的权重,然而对于在相机坐标系中不同深度的参考点和直线段,这样显然无法得到最高的精度。此外在很多情况下个别的异常参考点(包括参考直线段的端点)会有较大的测量误差,在实际应用中这种难以避免的误差会较大地影响位姿估计算法的计算精度。因此可以对不同的参考点和直线段加入不同的权重系数,以提高计算精度避免异常点的影响。加入权重系数以后的目标函数如下:

$$E(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^n \omega_i \|(I - V_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})\|^2 + \sum_{i=1}^m \sum_{j=1}^2 \omega_j \|(I - U_i)(\mathbf{R}\mathbf{p}_j^{l_i} + \mathbf{t})\|^2$$

$$= \sum_{i=1}^N \omega_i \|(I - W_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})\|^2 \quad (8)$$

式中: $\omega_j > 0$ 为第 j 个参考点(包括直线段的两个端点)的权重系数。加入权重后,当给定第 k 次对于 $\mathbf{R}$ 的估计值 $\mathbf{R}^{(k)}$ 以后,可以得出此时 $\mathbf{t}$ 的最优值为:

$$\mathbf{t}(\mathbf{R}^{(k)}) = \left(\sum_{i=1}^N \omega_i (I - W_i) \right)^{-1} \sum_{i=1}^N \omega_i (W_i - I) \mathbf{R}^{(k)} \mathbf{p}_i \quad (9)$$

同样设 $\mathbf{q}_i^{(k)} = V_i(\mathbf{R}^{(k)} \mathbf{p}_i + \mathbf{t}^{(k)})$, 由此可以得到第 k + 1 次对 $\mathbf{R}$ 的估计值 $\mathbf{R}^{(k+1)}$ 为:

$$\mathbf{R}^{(k+1)} = \arg \min_{\mathbf{R}} \sum_{i=1}^N \omega_i \left\| \mathbf{R} \mathbf{p}_i + \mathbf{t} - \mathbf{q}_i^{(k)} \right\|^2 \quad (10)$$

对于式(10)可以采用奇异值分解[10]求出 $\mathbf{R}^{(k+1)}$,

定义 $\bar{\mathbf{p}} = \sum_{i=1}^N \omega_i \mathbf{p}_i / \sum_{i=1}^N \omega_i$ 、 $\bar{\mathbf{q}}^{(k)} = \sum_{i=1}^N \omega_i \mathbf{q}_i^{(k)} / \sum_{i=1}^N \omega_i$

为参考点和其视线投影点的加权均值,定义

$\mathbf{p}_i' = \mathbf{p}_i - \bar{\mathbf{p}}$ 、 $\mathbf{q}_i'^{(k)} = \mathbf{q}_i^{(k)} - \bar{\mathbf{q}}^{(k)}$, 定义矩阵 $\mathbf{M} = \sum_{i=1}^N \omega_i \mathbf{q}_i'^{(k)} \mathbf{p}_i'^T$,

对 $\mathbf{M}$ 进行 SVD 分解得到 $\mathbf{M} = \mathbf{U} \mathbf{D} \mathbf{V}^T$, 则可以得到当 $\mathbf{R}^{(k+1)} = \mathbf{U} \mathbf{V}^T$ 时, 式(10)取最小值。从 $\mathbf{R}$ 的初始值开始

进行迭代直到满足停止条件, 则可以得到旋转矩阵 $\mathbf{R}$ 和平移向量 $\mathbf{t}$ 。文献[13]证明了该算法的全局收敛性。

1.2.2 权重系数确定方法

权值由以下方式获得: 首先使用 OPnP^[14] 算法 (Optimal Perspective-n-Point) 或者 RPnP^[15] 算法 (Robust Perspective-n-Point) 求得加权扩展迭代的初始估计值 $\mathbf{R}_{\text{init}}$ 、 $\mathbf{T}_{\text{init}}$, 由实验经验可知当参考点数目较少时, 使用 OPnP 算法的计算值作为扩展正交迭代算法的初始值时, 收敛效果较差, 使用 RPnP 算法的计算值则效果较好, 当参考点数目较多的时候则相反。所以计算初始值时首先判断参考点数目, 再根据参考点数目选择 OPnP 算法或者 RPnP 算法计算初值。得到初始值后可以使用初始值得到参考点和参考直线端点在相机坐标系下的坐标 $\mathbf{q}_{\text{init}}^i(x, y, z) = \mathbf{R}_{\text{init}} \mathbf{p}_i + \mathbf{T}_{\text{init}}$, 并根据公式(11)计算每个参考点和直线端点的重投影误差 r_i :

$$r_i = \left\| \mathbf{p}_i - \frac{1}{\mathbf{q}_{\text{init}}^i(z)} \mathbf{K}(\mathbf{R}_{\text{init}} \mathbf{p}_i + \mathbf{T}_{\text{init}}) \right\|^2 \quad (11)$$

式中: $\mathbf{K}$ 为相机内参矩阵。

令参考点的权值 D_1^i 为 $\mathbf{q}_{\text{init}}^i(x, y, z)$ 的深度的倒数的平方即: $D_1^i = 1 / \mathbf{q}_{\text{init}}^i(z)^2$ 。根据投影点的重投影误差 r_i 来确定权值 D_2^i :

$$D_2^i = \begin{cases} 1, & r_i^2 \leq r \\ r / r_i^2, & r_i^2 \geq r \end{cases} \quad (12)$$

式中: r 为 r_i^2 的均值。由此可以得到既考虑了参考点深度又考虑了初始值的重投影误差的权值:

$$\omega_i = D_1^i D_2^i \quad (13)$$

1.3 快速加权扩展正交迭代

正交迭代算法作为非线性 PnP 算法计算时间较长, 在对每个参考点加入权值以及扩展到直线段以后又增加了较大的计算量, 计算消耗时间进一步增加, 本文通过将迭代算法公式中的矩阵拉直, 并通过整合使加权扩展正交迭代算法每次迭代的计算复杂度从

$O(n)$ 降到了 $O(1)$, 具体推导过程如下:

首先引入公式:

$$\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\text{vec}(\mathbf{B}) \quad (14)$$

式中: vec 表示以列为单位把一个矩阵排为一个列向量, 称为矩阵的拉直; $\otimes$ 表示克罗内克积 (Kronecker product)。对式(9)做拉直运算得:

$$\begin{aligned} \text{vec}(\mathbf{t}(\mathbf{R}^{(k)})) &= \mathbf{t}(\mathbf{R}^{(k)}) = \left(\sum_{i=1}^N \omega_i (\mathbf{I} - \mathbf{W}_i) \right)^{-1} \sum_{i=1}^N \omega_i \mathbf{p}_i^T \otimes \\ &(\mathbf{W}_i - \mathbf{I}) \text{vec}(\mathbf{R}^{(k)}) = \mathbf{G}_{3 \times 9} \mathbf{r} \end{aligned} \quad (15)$$

式中:

$$\mathbf{r} = \text{vec}(\mathbf{R}^{(k)}) \quad (16)$$

$$\mathbf{G} = \left(\sum_{i=1}^N \omega_i (\mathbf{I} - \mathbf{W}_i) \right)^{-1} \sum_{i=1}^N \omega_i \mathbf{p}_i^T \otimes (\mathbf{W}_i - \mathbf{I}) \quad (17)$$

对 $\mathbf{q}_i^{(k)}$ 做拉直运算得:

$$\text{vec}(\mathbf{q}_i^{(k)}) = \mathbf{q}_i^{(k)} = \mathbf{W}_i (\mathbf{R}^{(k)} \mathbf{p}_i + \mathbf{t}^{(k)}) = (\mathbf{p}_i^T \otimes \mathbf{W}_i + \mathbf{W}_i \mathbf{G}) \mathbf{r} \quad (18)$$

进一步可得:

$$\text{vec} \left(\sum_{i=1}^N \omega_i \mathbf{q}_i \right) = \sum_{i=1}^N \omega_i \mathbf{q}_i = \sum_{i=1}^N \omega_i (\mathbf{p}_i^T \otimes \mathbf{W}_i + \mathbf{W}_i \mathbf{G}) \mathbf{r} = \mathbf{A} \mathbf{r} \quad (19)$$

式中:

$$\mathbf{A} = \sum_{i=1}^N \omega_i (\mathbf{p}_i^T \otimes \mathbf{W}_i + \mathbf{W}_i \mathbf{G}) \quad (20)$$

整理 $\mathbf{M}$ 公式得:

$$\mathbf{M} = \sum_{i=1}^N \omega_i \mathbf{q}_i^{(k)} \mathbf{p}_i^{(T)} = \left(\sum_{i=1}^N \omega_i \mathbf{q}_i \mathbf{p}_i^T \right) - \frac{1}{\sum_{i=1}^N \omega_i} \left(\sum_{i=1}^N \omega_i \mathbf{q}_i \right) \left(\sum_{i=1}^N \omega_i \mathbf{p}_i \right)^T \quad (21)$$

对 $\mathbf{M}$ 进行拉直运算可以得到:

$$\begin{aligned} \text{vec}(\mathbf{M}) = \mathbf{m} &= \left(\sum_{i=1}^N \omega_i \mathbf{p}_i \otimes (\mathbf{p}_i^T \otimes \mathbf{W}_i + \mathbf{W}_i \mathbf{G}) \right) - \\ &\frac{1}{\sum_{i=1}^N \omega_i} \left(\sum_{i=1}^N \omega_i \mathbf{p}_i \right) \otimes \mathbf{A} \mathbf{r} = \mathbf{B}_{9 \times 9} \mathbf{r} \end{aligned} \quad (22)$$

式中:

$$\mathbf{B} = \sum_{i=1}^N \omega_i \mathbf{p}_i \otimes (\mathbf{p}_i^T \otimes \mathbf{W}_i + \mathbf{W}_i \mathbf{G}) - \frac{1}{\sum_{i=1}^N \omega_i} \left(\sum_{i=1}^N \omega_i \mathbf{p}_i \right) \otimes \mathbf{A} \quad (23)$$

因此可以得到 $\mathbf{M}$, 对 $\mathbf{M}$ 进行奇异值分解 (SVD) 即可得到 $\mathbf{R}^{(k+1)}$ 。

在每次迭代过程中对误差函数进行规整可得:

$$\begin{aligned} E(\mathbf{R}, \mathbf{t}) &= \sum_{i=1}^N \left\| \omega_i (\mathbf{I} - \mathbf{W}_i) (\mathbf{R} \mathbf{p}_i + \mathbf{t}) \right\|^2 = \\ &\sum_{i=1}^N \left\| \omega_i (\mathbf{I} - \mathbf{W}_i) (\mathbf{p}_i^T \otimes \mathbf{I} + \mathbf{G}) \mathbf{r} \right\|^2 = \mathbf{r}^T \mathbf{C}_{9 \times 9} \mathbf{r} \end{aligned} \quad (24)$$

式中:

$$\mathbf{C} = \sum_{i=1}^N \omega_i^2 (\mathbf{p}_i \otimes \mathbf{I} + \mathbf{G}^T) (\mathbf{I} - \mathbf{W}_i) (\mathbf{p}_i^T \otimes \mathbf{I} + \mathbf{G}) \quad (25)$$

经过对 $\mathbf{t}$ 、 $\mathbf{M}$ 和 $E(\mathbf{R}, \mathbf{t})$ 进行拉直运算后, 将每次迭代的变量 $\mathbf{R}$ 提取出来, 可以在迭代前只计算一次 $\mathbf{G}$ 、 $\mathbf{B}$ 和 $\mathbf{C}$, 计算复杂度均为 $O(n)$, 在迭代时只需与拉直后的 $\mathbf{R}$ 相乘, 计算复杂度为 $O(1)$, 从而可以极大地减少加权扩展正交迭代算法的计算时间, 增强算法的实时性。

2 实验

为了验证本文提出的快速加权的扩展正交迭代算法的正确性和有效性, 我们进行了相机位姿估计的 matlab 数值仿真实验和实物实验, 分别比较以下 9 种算法的计算误差和消耗时间: (1) RPnP: [15]提出的 RPnP 算法; (2) OPnP: [14]提出的 OPnP 算法; (3) LHM: [10]提出的以弱透视模型 (Weak Perspective Approximation) 为初值的正交迭代算法; (4) RPnP/OPnP+LHM: 使用 RPnP 和 OPnP 算法按照 1.2.2 小节的方式计算迭代算法的初值, 然后使用正交迭代算法; (5) RPnP/OPnP+ELHM: 使用上述初值, 然后使用[12]提出的扩展正交迭代算法; (6) RPnP/OPnP+D1+ELHM: 使用上述初值, 然后使用只考虑深度信息的加权扩展正交迭代; (7) RPnP/OPnP+D2+ELHM: 使用上述初值, 然后使用只考虑投影误差的加权扩展正交迭代; (8) RPnP/OPnP+W+ELHM: 使用上述初值, 然后使用考虑两种权重系数的加权扩展正交迭代; (9) RPnP/OPnP+W+ELHM+: 使用上述初值, 然后使用经过加速运算的快速加权扩展正交迭代。

2.1 仿真实验

仿真实验使用的虚拟相机的参数: 分辨率为 640×480 , 等效焦距为 800 pixel, 参考点和参考直线段的端点随机分布在 $[-2, 2] \times [-2, 2] \times [4, 8]$ 的空间中。仿真实验中每次随机产生 3 个欧拉角并将其转换成旋转矩阵 $\mathbf{R}$, 在 $[-50, 50] \times [-50, 50] \times [0, 100]$ 的范围内随机产生平移向量 $\mathbf{t}$, 并将点投影在图像上后加入期望为 0, 标准差为 σ 的高斯噪声。对于旋转矩阵, 由于欧拉角的万向锁问题 (Gimbal Lock), 实验中将旋转矩阵转换成四元数来表示姿态, 令旋转矩阵真值 $\mathbf{R}$ 转换成四元数 $\mathbf{q}$, 计算得到的旋转矩阵转换 $\mathbf{R}'$ 成为四元数 $\mathbf{q}'$, 定义旋转矩阵的相对误差为 $e(\mathbf{R}) = (\mathbf{q} -$

$q')^2/q^2$ 。对于平移向量,令平移向量真值为 t ,计算得到的平移向量为 t' ,定义平移向量的相对误差为 $e(t) = (t - t')^2/t^2$ 。

2.1.1 参考点数目实验

参考点数目实验(研究存在异常参考点时算法计算精度和参考点数目关系):参考点数目从4增加到12,参考直线数目为2,随机一个参考点加上 σ 为5 pixel的高斯噪声,其余参考点加上 σ 为0.5 pixel的高斯噪声。对于不同的参考点数目都进行1000次独立重复试验。实验结果如图3所示。

从图3可以看到在有异常点和相同的初始值的情况下,加权扩展正交迭代算法对于旋转矩阵和平移向量的计算精度,相比于扩展正交迭代和正交迭代算法有较大提高并且保持在较高的水平。

2.1.2 异常点和高斯噪声实验

异常点和高斯噪声实验(研究存在异常参考点时算法计算精度和高斯噪声的关系):参考点数目固定为10,参考直线数目为2,高斯噪声标准差 σ 统一从1 pixel增加到10 pixel,每次随机选择一个参考点加上

标准差为当前标准差5倍的高斯噪声。对于不同的标准差 σ 都进行1000次独立重复试验。实验结果如图4所示。

从图4可以看到随着高斯噪声水平的增加,加权扩展正交迭代的计算精度始终最高,相比于没有考虑加权系数的扩展正交迭代有较大提高。同时可以看出只考虑深度信息的加权扩展正交迭代计算精度高于没有加权的扩展正交迭代,只考虑初始值的重投影误差的扩展正交迭代对异常点的抗干扰性较强。

2.1.3 平移距离实验

平移距离实验(研究存在异常点和高斯噪声时计算精度和平移距离的关系):固定参考点数目为10,参考直线数目为2,随机一个参考点加上 σ 为10 pixel的高斯噪声,其余参考点加上 σ 为1 pixel的高斯噪声,令平移向量 t_x 和 t_y 在 $[-20, 20]$ (步长5乘上参考点区域大小4)的区域内随机产生,令参考点区域到相机距离与参考点区域大小之比 $t_z/4$ 以步长为5从5增加到40,每次做1000次独立重复性实验,实验结果如图5所示。

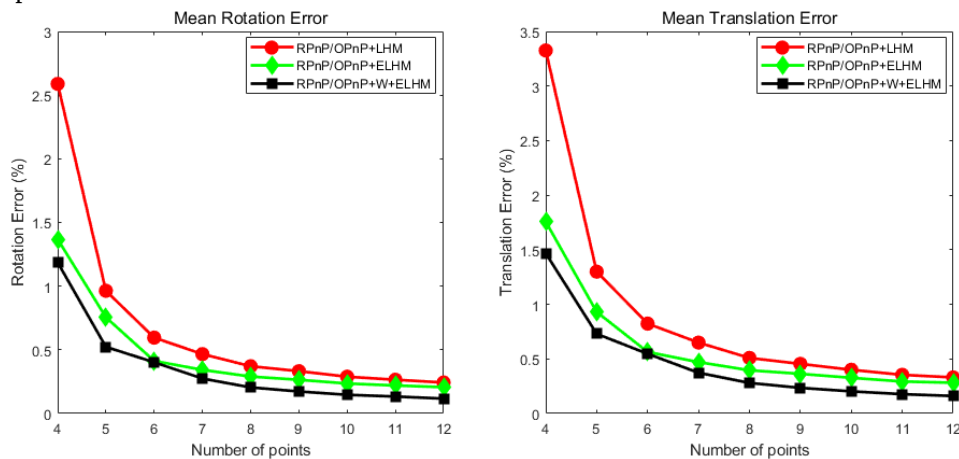


图3 参考点数目增加时,旋转矩阵和平移向量的相对计算误差

Fig.3 Relative calculation error of the rotation matrix and translation vector when the number of reference points increases

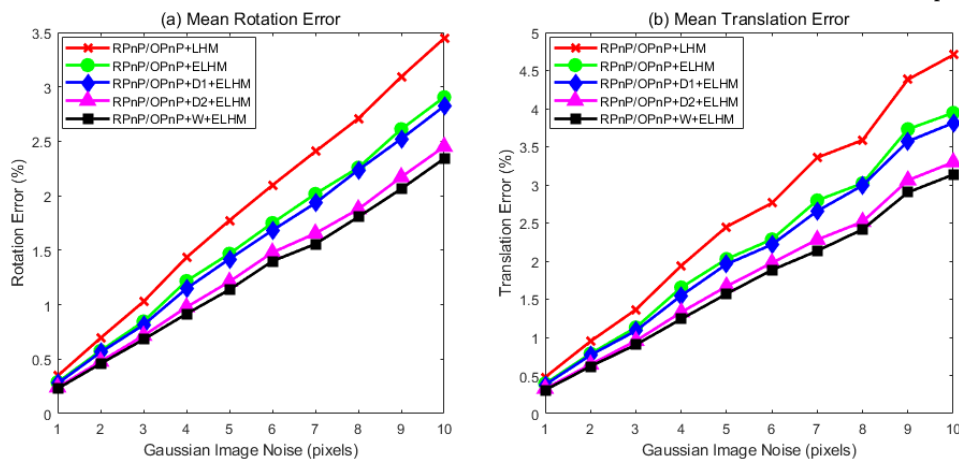


图4 高斯噪声水平增加时,旋转矩阵和平移向量的相对计算误差

Fig.4 Relative computational error of the rotation matrix and translation vector as the Gaussian noise level increases

从图 5 可以看到随着距离的增加, 加权扩展正交迭代的计算精度始终最高, 且受距离的影响较小。

2.1.4 计算时间实验

计算时间实验 (研究算法的计算速度随参考点数目变化的变化): 参考点数目从 4 增加到 104, 参考直线数目为 2, 在参考点上加上标准差 σ 为 3pixel 的高斯噪声, 计算 1000 次独立重复试验算法消耗时间的平均值, 实验结果如图 6 所示。

从图 6 可以看出, 没有加速的加权扩展正交迭代的计算时间远多于扩展正交迭代和正交迭代算法, 而加速后的加权扩展正交迭代算法则大量节省了计算时间, 大约可以减少 40 ms。在参考点增多以后, 因为需要对每个参考点和参考直线段端点加以权重的原因, 计算时间会多于没有加速的正交迭代算法, 但是少于没有加速的加权正交迭代。

2.2 实物实验

在真实图像实验中, 使用 KITTI^[16] (Karlsruhe Institute of Technology and Toyota Technological Institute at Chicago) 标准数据集对 PnP 算法进行验证。KITTI 数据集是由德国卡尔斯鲁勒理工学院搭建, 包括了城市、乡村、高速公路等多种环境, 通过自动驾驶汽车进行采集, 车上载有多种传感器, 其中高精度的组合导航系统为数据集提供相机位姿真值。数据集中的图片分辨率为 1226 pixel $\times$ 370 pixel, 相机等效焦距为 707.0912 pixel。

在实验中首先对序列图片进行特征点匹配和直线段匹配, 直线段检测使用 LSD^[17] 算法 (line segment detector), 得到特征点和直线段端点的坐标, 其中直线匹配结果如图 7 所示。

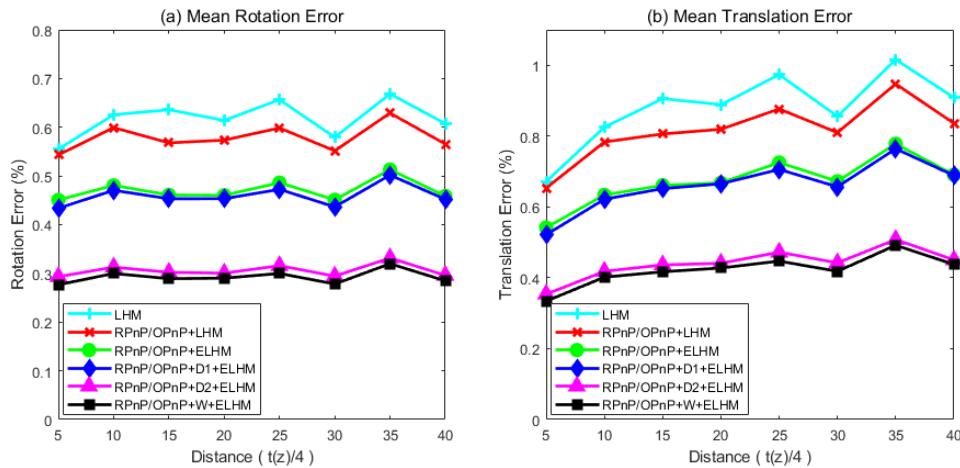


图 5 相机与参考点区域距离增加时, 旋转矩阵和平移向量的相对计算误差

Fig.5 Relative calculation error of the rotation matrix and translation vector as the distance between the camera and the reference point area increases

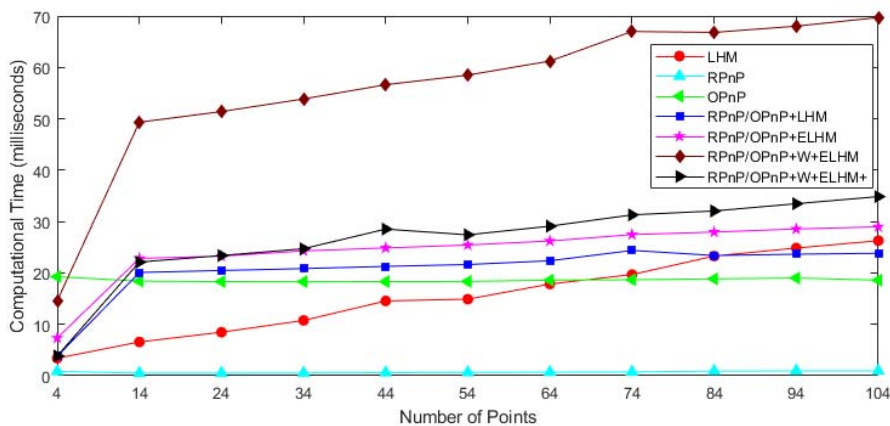


图 6 参考点数目增加时, 各种算法的计算时间对比

Fig.6 Comparison of calculation time of various algorithms when the number of reference points increases

初始化时通过对极几何计算第一帧和第二帧之间的旋转和平移矩阵,通过三角化计算匹配的特征点和直线端点的三维坐标,然后利用三维坐标和特征点和直线段点在下一帧的二维坐标计算相机的旋转和平移。初始化后则直接利用上一次 PnP 算法计算的旋转和平移矩阵,通过三角化来计算三维坐标以进行下一次的 PnP。实验结果如图 8, 9 所示。

从图 8, 9 中可以看到在序列图片实验中本文提

出算法的计算误差远小于传统正交迭代算法的误差,计算准确性较高。由于直线提取以及匹配的不稳定性,在个别图像中加速加权扩展正交迭代算法的计算误差大于传统正交迭代算法,但是仍保持在较低水平,并且这种情况出现的次数较少,不影响整体计算的准确性。在序列图像中的实验证明了提出的算法有较高的准确性和鲁棒性,充分利用了图像中的信息,在实际场景应用中有较高价值。

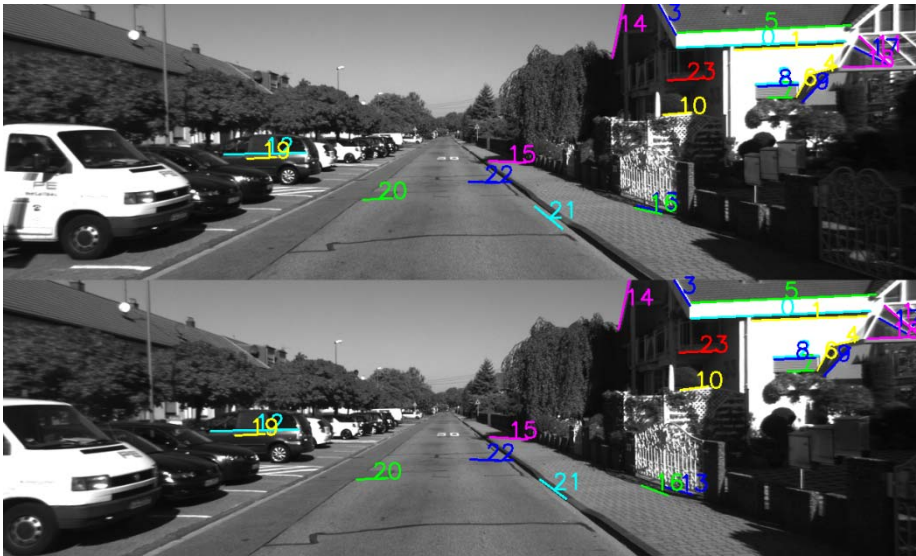


图 7 序列图像直线匹配结果,相匹配的直线用同样的颜色和相同的序号标注

Fig.7 The result of line matching in sequence images, and the matched lines are marked with the same color and the same serial number

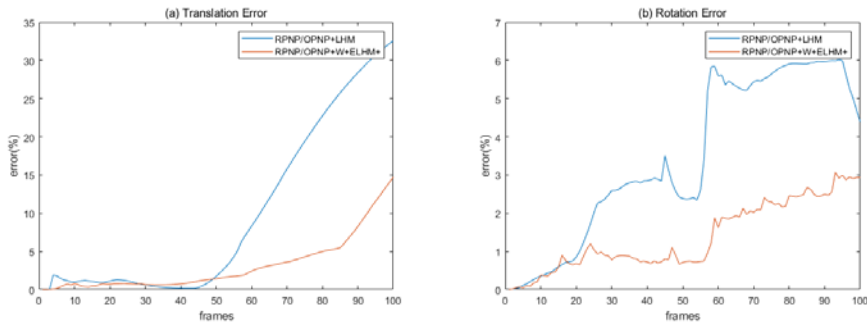


图 8 序列 5 中旋转矩阵和平移向量的计算误差

Fig.8 Calculation error of rotation matrix and translation vector in sequence 5

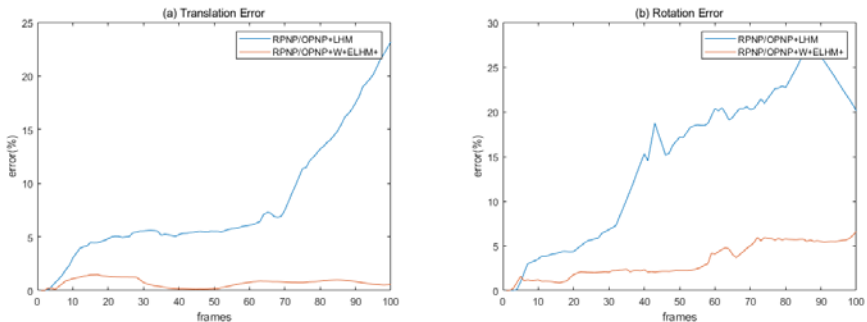


图 9 序列 6 中旋转矩阵和平移向量的计算误差

Fig.9 Calculation error of rotation matrix and translation vector in sequence 6

由于在序列图片中每次进行位姿估计时,需要根据上一次的计算结果来使用 PnP 算法计算本次的位姿,使用快速加权扩展正交迭代算法会有一定的累积误差,但是由于提取了序列图片中的直线信息并且对参考点和线段端点加以权重,该算法的累积误差远小于传统的正交迭代算法的累积误差。对于位姿计算实验中的累积误差,可以使用光束平差法^[5]对位姿进行优化从而抑制累积误差,同时可以使用回环检测^[5]的方法对位姿进行优化,从而在一定程度上消除累积误差。

3 结论

针对二维—三维特征对应(2D-3D)的相机姿态估计算法(PnP)进行了研究,在基于点和直线段的扩展正交迭代算法的基础上提出了快速加权扩展正交迭代算法。该算法可以利用图像中的直线信息,既考虑了参考点以及参考直线端点的深度信息,又考虑了参考点和参考直线端点的重投影误差,并通过公式推导将每一次迭代的时间复杂度从 $O(n)$ 降到了 $O(1)$,从而提高了算法的计算精度和计算速度。

数值仿真实验和真实图片实验,对所提出的位姿估计算法进行了验证,并与正交迭代算法和扩展正交迭代算法进行对比,结果证明该算法有效地提高了计算精度、计算速度,对异常点有较强的抗干扰能力,对复杂环境下的实时相机姿态估计问题具有一定价值。

参考文献:

- [1] Mullen R. *Manual of photogrammetry*[M]. Barlow: Asprs Publications, 2004.
- [2] Hartley R, Zisserman A. *Multiple view geometry in computer vision*[M]. Cambridge: Cambridge university press, 2003.
- [3] Gingold Y, Igarashi T, Zorin D. Structured annotations for 2D-to-3D modeling[C]//*ACM Transactions on Graphics (TOG)*. ACM, 2009, 28(5): 148.
- [4] DeSouza G N, Kak A C. Vision for mobile robot navigation: A survey[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24(2): 237-267.
- [5] Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. *IEEE Transactions on Robotics*, 2015, 31(5): 1147-1163.
- [6] 钱萍, 王惠南. 基于对偶四元数的航天器交会对接位姿双目视觉测量算法[J]. *宇航学报*, 2013, 34(1): 32-38.
- [7] QIAN Ping, WANG Huinan. A Binocular Vision Algorithm Based on Dual Quaternion for Relative Position and Attitude Determinati of RVD Spacecrafts[J]. *Journal of Astronautics*, 2013, 34(1): 32-38.
- [7] 吴凡路, 刘建军, 任鑫, 等. 基于圆形标志点的深空探测全景相机标定方法[J]. *光学学报*, 2013, 33(11): 1115002.
- [8] WU Fanlu, LIU Jianjun, REN Xin, et al. Deep space exploration panoramic camera calibration technique basedon circular markers[J]. *Acta Optica Sinica*, 2013, 33(11): 1115002.
- [8] Azuma R, Baillot Y, Behringer R, et al. Recent advances in augmented reality[J]. *IEEE computer graphics and applications*, 2001, 21(6): 34-47.
- [9] Haralick R M, Joo H, Lee C N, et al. Pose estimation from corresponding point data[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1989, 19(6): 1426-1446.
- [10] 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.
- [11] Pumarola A, Vakhitov A, Agudo A, et al. PL-SLAM: Real-time monocular visual SLAM with points and lines[C]//2017 *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017: 4503-4508.
- [12] 许允喜, 蒋云良, 陈方. 基于点和直线段对应的扩展正交迭代位姿估计算法[J]. *光学学报*, 2009(11): 3129-3135.
- [12] XU Yunxi, JIANG Yunliang, CHEN Fang. An Extended Orthogonal Iteration Algorithm for Pose Estimation from Point and Line Segment Correspondences [J]. *Acta Optica Sinica*, 2009, 29(11): 3129-3135.
- [13] 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.
- [13] ZHOU Run, ZHANG Zhengyu, HUANG Xuhui. Weighted Orthogonal Iteration Algorithm for Camera Pose Estimation[J]. *Acta Optica Sinica*, 2018, 38(5): 193-199.
- [14] ZHENG Y, KUANG Y, Sugimoto S, et al. Revisiting the pnp problem: A fast, general and optimal solution[C]//*Proceedings of the IEEE International Conference on Computer Vision*, 2013: 2344-2351.
- [15] LI S, 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.
- [16] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: The KITTI dataset[J]. *The International Journal of Robotics Research*, 2013, 32(11): 1231-1237.
- [17] VonGioi R G, Jakubowicz J, Morel J M, et al. LSD: A fast line segment detector with a false detection control[J]. *IEEE transactions on pattern analysis and machine intelligence*, 2008, 32(4): 722-732.