

文章编号: 0253-2239(2009)01-0072-06

多摄像机系统位姿估计的广义正交迭代算法

许允喜¹ 蒋云良^{1,2} 陈 方³

(¹ 湖州师范学院信息工程学院, 浙江 湖州 313000; ² 浙江大学计算机学院, 浙江 杭州 310027)
³ 南京航空航天大学 自动化学院, 江苏 南京 210016

摘要 目前,多摄像机系统的位姿估计还缺乏系统的方法,它通常通过求解透视 n 点问题或者求解使两组 3D 点集之间平方和误差最小的刚体变换来解决,这些方法都有局限性。正交迭代算法是基于点特征的单目视觉算法,快速且全局收敛,是目前性能最优的实时位姿估计算法之一,被广泛应用。提出了一种广义正交迭代算法把所有摄像机获取的全部图像作为整体计算得到相对位姿参数,是通用的多目视觉位姿估计算法。算法先把所有摄像机数据进行统一表达,再把所有摄像机观测到的全部特征点的目标空间共线性误差平方和作为误差函数,最后经数学推导得到使该误差函数最小化的迭代求解过程。实验结果验证了算法的有效性以及在多摄像机系统位姿估计中的优越性。

关键词 机器视觉; 广义正交迭代算法; 位姿估计; 多摄像机系统; 交会对接

中图分类号 TP391.41 **文献标识码** A **doi**: 10.3788/AOS20092901.0072

Generalized Orthogonal Iterative Algorithm for Pose Estimation of Multiple Camera Systems

Xu Yunxi¹ Jiang Yunliang^{1,2} Chen Fang³

(¹ Institute of Information Engineering, Huzhou Teachers College, Huzhou, Zhejiang 31300, China)
² College of Computer Science, Zhejiang University, Hangzhou, Zhejiang 310027, China
³ College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, China

Abstract It still lacks of systematic method for solving pose estimation of a multiple camera system (MCS). The pose-estimation problem is usually achieved by solving the Perspective- n -point problem or finding the least-squared-error rigid transformation between two 3D point sets. They have limitations. The orthogonal iteration (OI) algorithm is characteristic can converge fast and globally is based on points. And it is one of the state-of-art real-time pose estimation algorithms and used widely. A generalized iterative algorithm is proposed. It is a universal multi-view algorithm for pose estimation. In the method, all images acquired from the cameras are treated as a whole, and the relative pose parameter is calculated this method, datum of multiple cameras is unified, and the sum of object-space collinearity error of feature points all cameras sensed is used as error function for MCS. An iterative algorithm is developed to minimize the error function. The experiment result demonstrates its effectiveness and superiority of the algorithm for pose estimation of MCS.

Key words machine vision; generalized orthogonal iteration algorithm; pose estimation; multiple camera system; rendezvous and docking

1 引 言

位姿估计问题是摄影测量学、计算机视觉、计算机图形学及机器人等学科的一个重要的基本问题,是航天器交会对接视觉测量系统^[1]、移动机器人导航与定位、自主航天器入港、机器人手眼定标等许多

工程实践和理论研究所要解决的核心问题。

已有众多学者对位姿估计算法进行了研究,但目前几乎所有的位姿估计算法都是单目视觉算法,不是专门针对多摄像机系统的,因而不能充分地利用多摄像机系统提供的全部信息。本文对目前

收稿日期: 2008-09-17; 收到修改稿日期: 2008-11-27

基金项目: 国家自然科学基金(60872057)和浙江省自然科学基金(Y1080212)资助课题。

作者简介: 许允喜(1978—),男,硕士,讲师,主要从事机器视觉、视觉导航方面的研究。E-mail: xuyunxi@tom.com

被广泛应用的基于点特征的正交迭代实时位姿估计算法进行了研究,在正交迭代算法基础上推导出针对多摄像机系统的广义正交迭代算法。该算法利用了多摄像机系统获取的全部图像,是通用的多目视觉位姿估计算法。算法把多摄像机系统观测到的所有特征点的目标空间共线性误差平方和作为误差函数,经数学推导得到使该误差函数最小化的迭代求解过程。最后,通过实验,并与正交迭代算法和最优绝对定向解法进行对比,验证了广义正交迭代算法在多摄像机系统位姿估计中的优越性。

2 位姿估计算法研究概况

位姿估计算法一般分为二类:线性解法和非线性解法。对于 3 个或 4 个非共线的特征点,可以采用线性算法得到位姿闭环解,由于在实际应用成像过程中的测量误差、量化误差、特征点提取误差或特征点匹配错误等因素影响,使用闭环解算法可能会产生很大的误差。

对 4 个以上的特征点,一般采用非线性优化算法^[2,3];然而多数的非线性优化方法需要先由线性算法求出一个初始解,才能收敛到正确解,且计算过程较复杂,系统实时性受到影响。Levenberg-Marquardt 方法(高斯-牛顿方法的变形)是解决非线性最小平方问题的经典方法。

其后,文献[4]提出了一种位姿估计算法:正交迭代(Orthogonal iteration, OI)算法,它是求解透视 n 点问题的单目视觉算法,采用迭代求解方法,具有迭代次数少,计算精度高,计算速度快且全局收敛的特点,可用于实时位姿估计。算法达到了与 Levenberg-Marquardt 方法相当的精确度,但计算效率更高,受初始值影响很小,对错误数据的鲁棒性更高。

之后,文献[5]给出了基于点或线特征的位姿估计方法,该方法是不要求初始化的线性解法。文中把其与正交迭代算法作了对比,计算精度和实时性方面都不及正交迭代算法。文献[6]采用了遗传算法用于增强现实系统的位姿估计,算法鲁棒性较高,但计算时间长,不能用于实时系统。文献[7]提出了相对于共面目标的鲁棒位姿估计算法,但运行时间太长,也不能用于实时计算。

事实上,Levenberg-Marquardt 方法和正交迭代算法已经成为目前性能最优的实时位姿估计算法,被广泛应用^[8~10]。

近年来多摄像机视图几何及其计算方法在计算

机视觉中受到很多关注,多摄像机系统(multiple camera systems, MCS)成为研究热点。在视觉系统中使用更多的摄像机,既可以利用多视角得到物体的完整信息,也可以利用冗余的信息,以换取系统对噪声的鲁棒性^[11,12]。

单目视觉算法用于多摄像机系统位姿估计求解旋转矩阵和平移向量仅使用单幅图像,无法实现对多摄像机获取的所有图像进行全局、统一计算。文献[13]采用多个摄像机,先对每个摄像机获取的单幅图像分别计算得到位姿估计参数,最后对结果进行投票获得最后的位姿估计参数,以增加鲁棒性。

用于多摄像机系统位姿估计的另一种方法就是求解使二组 3D 点集之间平方和误差最小的刚体变换,这种方法在计算机视觉领域已研究得比较成熟。由多摄像机系统的多视立体视觉重建 3D 摄像机坐标 q_i 。由三个或三个以上的非共线目标参考点 p_i , R 和 t 可以由下列最小平方和的解确定

$$\min_{R, t} \sum_i \|Rp_i + t - q_i\|^2, \quad (1)$$

约束条件为 $R^T R = I$ 。

上述受约束的最小平方和问题可以用闭环形式的最优绝对定向解法进行求解^[14,15]。用上述方法解决多摄像机系统位姿估计问题主要有 2 点局限性:在重建 3D 点 q_i 时,只被一个摄像机观测到的目标特征点不能被利用;算法在高噪声下 3D 重建误差很大,因而位姿估计误差也很大。

目前,对多摄像机系统的位姿估计还缺乏系统的方法。本文在正交迭代算法基础上推导出针对多摄像机系统的广义正交迭代算法,所有摄像机获取的任何特征点都能被利用并在算法中进行全局、统一计算,是通用的多目视觉位姿估计算法。

3 正交迭代算法

3.1 摄像机目标空间共线性误差

给定目标 3D 特征点坐标 $p_i = (x_i, y_i, z_i)^T$, 在摄像机坐标系下的对应点为 $q_i = (x'_i, y'_i, z'_i)^T$, 二者变换关系为

$$q_i = Rp_i + t, \quad (2)$$

式中 $R = (r'_1, r'_2, r'_3)^T$, $t = (t_x, t_y, t_z)^T$ 分别为旋转矩阵和平移向量。

如图 1 所示, p_i 投影在归一化图像平面得到的像点为 $v_i = (u_i, v_i, 1)^T$ 。目标空间中的共线性方程^[3]为

$$Rp_i + t = V_i(Rp_i + t), \quad (3)$$

则目标空间共线性误差^[4]为

$$e_i = (\mathbf{I} - \mathbf{V}_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t}), \quad (4)$$

式中 $\mathbf{V}_i = \mathbf{v}_i\mathbf{v}_i^T / (\mathbf{v}_i^T\mathbf{v}_i)$ 为沿视线方向的投影矩阵。

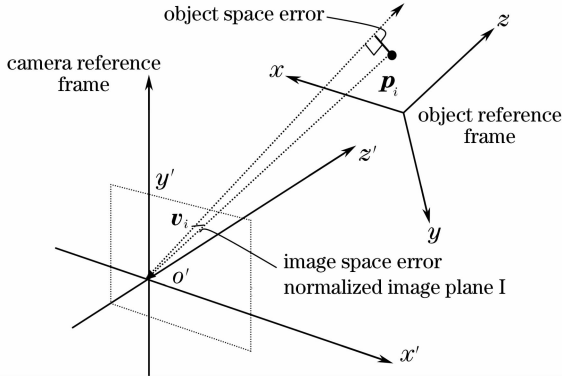


图 1 位姿估计的目标空间共线性误差

Fig. 1 Object-space collinearity errors for pose estimation

3.2 算法

正交迭代算法就是求解 \mathbf{R} 和 \mathbf{t} 使得共线性平方和误差最小。取目标函数

$$E(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^n \|e_i\|^2 = \sum_{i=1}^n \|(\mathbf{I} - \mathbf{V}_i)(\mathbf{R}\mathbf{p}_i + \mathbf{t})\|^2, \quad (5)$$

约束条件为 $\mathbf{R}^T\mathbf{R} = \mathbf{I}$ 。

当旋转矩阵 \mathbf{R} 给定时,可得到 \mathbf{t} 关于 \mathbf{R} 的最优解^[4]

$$\mathbf{t}(\mathbf{R}) = \frac{1}{n} \left(\mathbf{I} - \frac{1}{n} \sum_i \mathbf{V}_i \right)^{-1} \sum_i (\mathbf{V}_i - \mathbf{I})\mathbf{R}\mathbf{p}_i. \quad (6)$$

定义

$$q_i(\mathbf{R}) = \mathbf{V}_i[\mathbf{R}\mathbf{p}_i + \mathbf{t}(\mathbf{R})], \quad (7)$$

$$\bar{\mathbf{p}} = \frac{1}{n} \sum_{i=1}^n \mathbf{p}_i,$$

$$\bar{\mathbf{q}}(\mathbf{R}) = \frac{1}{n} \sum_{i=1}^n q_i(\mathbf{R}),$$

$$\mathbf{M}(\mathbf{R}) = \sum_{i=1}^n q'_i(\mathbf{R})\mathbf{p}'_i,$$

其中

$$\mathbf{p}'_i = \mathbf{p}_i - \bar{\mathbf{p}}, \quad (8)$$

$$q'_i(\mathbf{R}) = q_i(\mathbf{R}) - \bar{\mathbf{q}}(\mathbf{R}),$$

迭代求解过程为:首先假设 \mathbf{R} 的第 k 次估计为 $\mathbf{R}^{(k)}$, 则得 $\mathbf{t}^{(k)} = \mathbf{t}[\mathbf{R}^{(k)}]$, $q_i^{(k)} = \mathbf{R}^{(k)}\mathbf{p}_i + \mathbf{t}^{(k)}$ 。则

$$\mathbf{R}^{(k+1)} = \arg \min_{\mathbf{R}} \sum_{i=1}^n \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{V}_i q_i^{(k)}\|^2 = \arg \min_{\mathbf{R}} \sum_{i=1}^n \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - q_i[\mathbf{R}^{(k)}]\|^2 \quad (9)$$

$$= \arg \max_{\mathbf{R}} \text{tr}[\mathbf{R}^T \mathbf{M}(\mathbf{R}^{(k)})], \quad (10)$$

(10)式可以使用奇异值分解(SVD)^[4]来求解。

重复上述过程。

4 多摄像机系统位姿估计的广义正交迭代算法

正交迭代算法是单目视觉算法,其计算公式是针对单个摄像机。为了最优地利用多摄像机数据,要把全部图像进行统一计算得到相对位姿参数。为了达到这个目标,先从数据的几何关系上对用于计算的多摄像机数据进行统一表达,然后把多摄像机系统获取的全部图像特征点看作“单个广义摄像机”获取的图像特征点,建立其目标误差函数,最后进行算法的推导。

4.1 多摄像机数据的统一表达

设多摄像机系统有 M 个摄像机(如图 2)。 $x-y-z$ 为目标参考坐标系,特征点在该坐标系下设为 \mathbf{p}_i 。 $x_j-y_j-z_j$ ($j=1,2,\dots,M$) 为各摄像机的参考坐标系, o_j 为摄像机 j 的光心,特征点 \mathbf{p}_i 相对于摄像机 j 坐标系设为 q_i^j ($j=1,2,\dots,M$)。 \mathbf{p}_i 投影在摄像机 j 归一化图像平面得到的像点坐标为 $\mathbf{v}_i^j = (u_i^j, v_i^j, 1)^T$ ($j=1,2,\dots,M$)。多摄像机系统的空间目标共线性误差如图。选定某摄像机 g 的坐标系为整个多摄像机系统的参考坐标系,摄像机 g 坐标系与目标参考坐标系之间的旋转矩阵、平移向量为待估计参数 \mathbf{R} 和 \mathbf{t} 。由于多摄像机系统中各个摄像机坐标都已标定,各个摄像机之间的相对位置和方向都是已知的,设其他的摄像机 j 坐标系与摄像机 g 坐标系之间的旋转矩阵为 \mathbf{R}^j , 平移向量为 \mathbf{t}^j 。

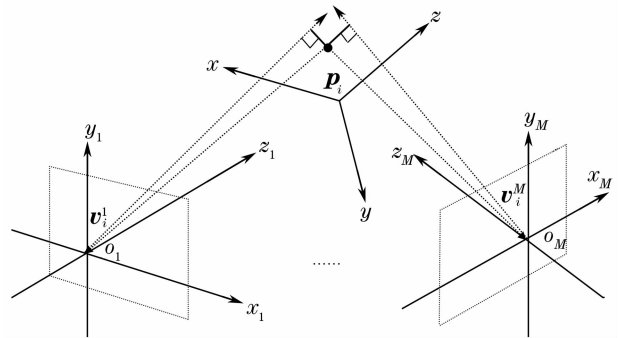


图 2 多摄像机系统位姿估计的目标空间误差

Fig. 2 Object-space errors of MCS for pose estimation

在正交迭代算法中,目标函数中的 q_i 、计算 \mathbf{V}_i 的 \mathbf{v}_i 事实上是向量 $o\mathbf{q}_i$ 和 $o\mathbf{v}_i$, 因为他们都是相对于同一个摄像机坐标系,其原点是光心 o , 所以点坐标和向量坐标是相同的,算法隐含表现了这一点。而在广义正交迭代算法中,各个摄像机得到的数据都相对于各自不同的摄像机坐标系,不能采用正交迭代算法相同的

处理方法。所以先把其他摄像机数据都转化在摄像机 g 坐标系下, 然后进行广义正交迭代算法的推导计算, 以下数据均相对于摄像机 g 坐标系。

目标物体的特征点 p_i 的坐标 $\mathbf{q}_i^g = \mathbf{R}p_i + \mathbf{t}$;

$\mathbf{o}_g \mathbf{q}_i^g$ 向量坐标 $\mathbf{q}_i^g = [\mathbf{q}_i^g - (0, 0, 0)^T] = \mathbf{q}_i^g$;

$\mathbf{o}_g \mathbf{v}_i^g$ 向量坐标 $\mathbf{v}_i^g = [\mathbf{v}_i^g - (0, 0, 0)^T] = \mathbf{v}_i^g$;

则沿 $\mathbf{o}_g \mathbf{v}_i^g$ 视线方向的投影矩阵

$$\mathbf{V}_i^g = \frac{\mathbf{v}_i^g \mathbf{v}_i^{gT}}{\mathbf{v}_i^{gT} \mathbf{v}_i^g} = \frac{\mathbf{v}_i^g \mathbf{v}_i^{gT}}{\mathbf{v}_i^{gT} \mathbf{v}_i^g}.$$

$\mathbf{o}_j \mathbf{q}_i^j$ 向量坐标

$$\mathbf{q}_i^j = \mathbf{q}_i^g - [\mathbf{R}_c^j * (0, 0, 0)^T + \mathbf{t}_c^j] = \mathbf{R}p_i + \mathbf{t} - \mathbf{t}_c^j, (j \neq g);$$

$\mathbf{o}_j \mathbf{v}_i^j$ 向量坐标

$$\mathbf{v}_i^j = (\mathbf{R}_c^j \mathbf{v}_i^j + \mathbf{t}_c^j) - [\mathbf{R}_c^j (0, 0, 0)^T + \mathbf{t}_c^j] = \mathbf{R}_c^j \mathbf{v}_i^j, (j \neq g);$$

则沿 $\mathbf{o}_j \mathbf{v}_i^j$ 视线方向的投影矩阵

$$\mathbf{V}_i^j = \frac{\mathbf{v}_i^j \mathbf{v}_i^{jT}}{\mathbf{v}_i^{jT} \mathbf{v}_i^j} = \frac{\mathbf{R}_c^j \mathbf{v}_i^j \mathbf{v}_i^{jT} \mathbf{R}_c^{jT}}{\mathbf{v}_i^{jT} \mathbf{R}_c^{jT} \mathbf{R}_c^j \mathbf{v}_i^j}, (j \neq g).$$

4.2 广义正交迭代算法

设目标共有 n 个特征点 p_i , A_j 为摄像机 j 观测到的特征点集, $|A_j|$ 为集合 A_j 中元素的个数。设多摄像系统观测到的特征点总数为 N , 则 $N = \sum_{j=1}^M |A_j|$,

$N \leq n * M$ 。设 $t^j = \begin{cases} 0, & (j = g) \\ t_c^j, & (j \neq g) \end{cases}$, 则

$$\mathbf{q}_i^j = \mathbf{R}p_i + \mathbf{t} - \mathbf{t}^j, (j = 1, \dots, M). \quad (11)$$

算法把所有摄像机观测到的全部特征点的目标空间共线性误差平方和作为多摄像机系统的目标误差函数, 则多摄像机系统位姿估计迭代求解过程如下:

1) 摄像机 j 获取的特征点 p_i 的目标空间共线性误差为

$$\mathbf{e}_i^j = (\mathbf{I} - \mathbf{V}_i^j) \mathbf{q}_i^j, \quad (12)$$

建立目标误差函数

$$E(\mathbf{R}, \mathbf{t}) = \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \|\mathbf{e}_i^j\|^2 = \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \|(\mathbf{I} - \mathbf{V}_i^j) \mathbf{q}_i^j\|^2 = \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \|(\mathbf{I} - \mathbf{V}_i^j)(\mathbf{R}p_i + \mathbf{t} - \mathbf{t}^j)\|^2, \quad (13)$$

该目标函数是所有摄像机观测到的所有特征点总的共线性误差。优化目标就是使该误差函数最小。单目视觉位姿估计算法是使单个摄像机的共线性误差最小。相对于单目视觉位姿估计算法而言, 目标是使所有单个摄像机的共线性误差之和最小, 是全局最小, 因此本方法计算得到的位姿估计更精确。

2) 给定 \mathbf{R} 的值, (11) 式的目标函数关于 \mathbf{t} 是二次方程, 经数学推导, \mathbf{t} 可以由闭环形式计算

$$\mathbf{t}^{(k)} = \mathbf{t}[\mathbf{R}^{(k)}] = \frac{1}{N} \left(\mathbf{I} - \frac{1}{N} \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \mathbf{V}_i^j \right)^{-1} \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} (\mathbf{V}_i^j - \mathbf{I}) [\mathbf{R}^{(k)} p_i - \mathbf{t}^j], \quad (14)$$

式中 k 表示第 k 次迭代。

3) 由 (11) 式得 $\mathbf{q}_i^{j(k)} = \mathbf{R}^{(k)} p_i + \mathbf{t}^{(k)} - \mathbf{t}^j$, 计算出 $\mathbf{q}_i^{j(k)}$ 。

4) 根据误差函数, 则

$$\mathbf{R}^{(k+1)} = \arg \min_{\mathbf{R}} \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \|\mathbf{R}p_i + \mathbf{t} - \mathbf{t}^j - \mathbf{V}_i^j \mathbf{q}_i^{j(k)}\|^2, \quad (15)$$

定义

$$\mathbf{Q}_i^j[\mathbf{R}^{(k)}] = [\mathbf{V}_i^j \mathbf{q}_i^{j(k)} + \mathbf{t}^j], \quad (16)$$

则

$$\mathbf{R}^{(k+1)} = \arg \min_{\mathbf{R}} \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \|\mathbf{R}p_i + \mathbf{t} - \mathbf{Q}_i^j[\mathbf{R}^{(k)}]\|^2, \quad (17)$$

(17) 式与正交迭代算法的 (9) 式形式类似。定义

$$\bar{\mathbf{p}} = \frac{1}{N} \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} p_i, \quad \bar{\mathbf{Q}}_i^j[\mathbf{R}^{(k)}] = \frac{1}{N} \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \mathbf{Q}_i^j[\mathbf{R}^{(k)}], \quad (18)$$

$$\mathbf{M}[\mathbf{R}^{(k)}] = \sum_{j=1}^M \sum_{i=1, p_i \in A_j}^{|A_j|} \mathbf{Q}_i^j[\mathbf{R}^{(k)}] p_i', \quad p_i' = p_i - \bar{\mathbf{p}}, \quad \mathbf{Q}_i^j[\mathbf{R}^{(k)}] = \mathbf{Q}_i^j[\mathbf{R}^{(k)}] - \bar{\mathbf{Q}}_i^j[\mathbf{R}^{(k)}], \quad (19)$$

所以 $\mathbf{R}^{(k+1)} = \operatorname{argmin}_{\mathbf{R}} \operatorname{tr}\{\mathbf{R}'\mathbf{M}[\mathbf{R}^{(k)}]\}$. (20)

5) 判断是否收敛,不收敛则返回步骤 2), 否则结束,输出位姿估计结果 \mathbf{R} 和 t 。

5 实验结果与分析

为了验证算法推导的正确性,采用正交迭代算法中的实验方法^[4],建立了 matlab 仿真实验系统,模拟计算位姿估计过程,这也是目前位姿估计算法最常用的研究手段^[1,4~10]:根据透视投影原理把物体投影到摄像机图像平面,由于摄像机误差和特征点提取误差,所以在图像特征点中再加入噪声。摄像机的等效焦距为 1600。本实验中,取 3 个摄像机组成多摄像机系统($M=3$)。由 $[-5,5] \times [-5,5] \times [-5,5]$ 空间定义的立方体的 8 个顶点作为目标特征点集。设三个摄像机获取到的特征点总数 $N=18$ 。 t 的相对误差为 $\|t' - t\| / \|t\|$, t' 为计算值, t 为真值; \mathbf{R} 的绝对误差为三个欧拉角的均方根误差 $\sqrt{(\theta' - \theta)^2 + (\phi' - \phi)^2 + (\gamma' - \gamma)^2}$, θ' , ϕ' , γ' 为计算值, θ , ϕ , γ 为真值。目标的特征点投影到归一化图像平面产生图像点,在图像点上加入零均值且标准方差为 σ 的高斯噪声。摄像距离表达为与目标大小

的比值 $t_z/10$ 。

在实验中,3 个旋转角 θ, φ, γ 按平均分布随机产生得到旋转矩阵 \mathbf{R} 。每个实验运行 1000 次,平均旋转误差和平移误差通过 1000 个均匀分布的旋转矩阵计算得到。为了说明广义正交迭代算法对多摄像机位姿估计的优越性,实验结果还与正交迭代算法和最优绝对定向解法进行了对比。正交迭代算法采用投票方式选择获取特征点最多的摄像机作为最后的位姿估计结果。

仿真实验 1(研究算法的抗噪声性能):固定摄像机距离 $t_z/10=10$, σ 从 1~5 pixel 变化。实验结果如图 3 所示,可以看出,在不同噪声水平下,广义正交迭代算法的位姿估计误差最小,抗噪声性能最优。例如,在 σ 为 5 pixel 时,广义正交迭代算法的角度计算误差仅为 1.5° ,而正交迭代算法和绝对定向解法的角度误差则分别为 2.5° 和 5.3° 。

仿真实验 2(研究算法与摄像距离之间的关系),图像噪声固定为 1 pixel,令 $t_z/10$ 为变量,以步长为 4,在 2~18 内变化,产生对应的一组 z 方向分量 t_z 。实验结果如图 4 所示,可以得出,在不同摄像距离下,广义正交迭代算法的位姿估计误差最小、最稳定。

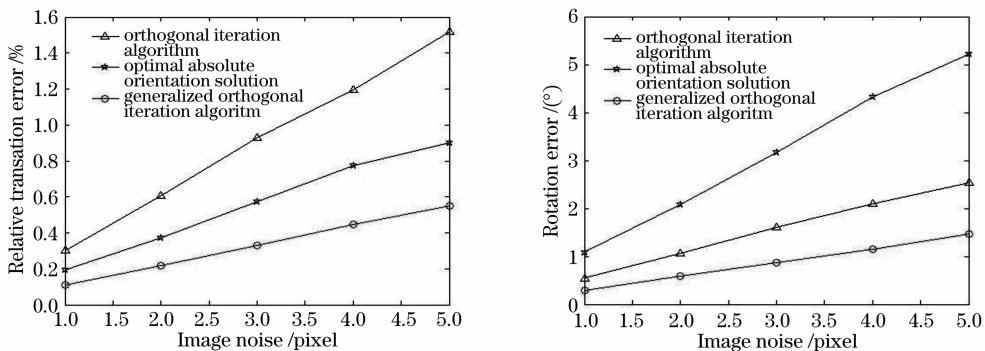


图 3 位姿估计误差与图像噪声的关系。(a)平移误差;(b)旋转误差

Fig. 3 Error of pose estimation as a function of image noise (a) translation error and (b) rotation error

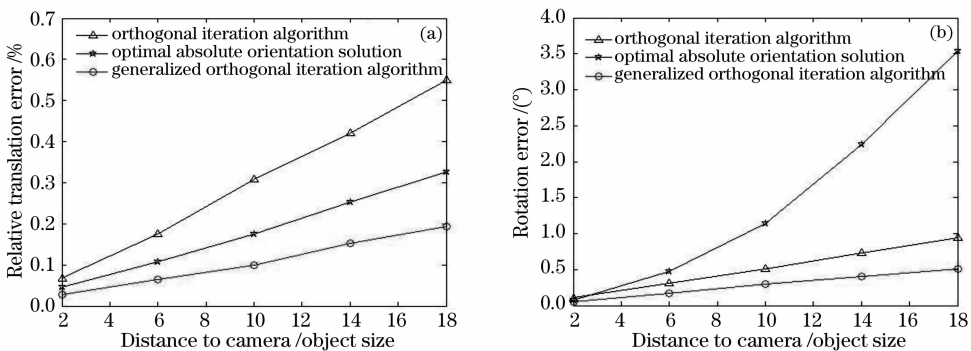


图 4 位姿估计误差与摄像距离的关系。(a)平移误差;(b)旋转误差

Fig. 4 Error of pose estimation as a function of distance to camera (a) translation error and (b) rotation error

仿真实验 3(研究算法的运行时间)。图像噪声固定为 1 pixel,令 $t_z/10=10$ 。平均运行时间如表 1 所示,可见三种算法都满足实时性要求。这是在 matlab 环境下的运行时间,在实际工程应用中运行时间将非常短。对于正交迭代算法,实验结果为单个摄像机的运行时间。虽然广义正交迭代算法使用三个摄像机的数据计算误差函数,但是广义正交迭代算法的运行时间远小于正交迭代算法的三倍。另外,图像噪声和摄像距离对运行时间影响很小,在不同噪声水平和摄像距离下得出的实验结果和表 1 差不多。

表 1 各种算法的运行时间

Table 1 Runtime of the algorithms

Algorithm	Runtime /s
Orthogonal iteration algorithm	0.027
Optimal absolute orientation solution	0.003
Generalized orthogonal iteration algorithm	0.037

由以上三个实验得出,三个算法都满足实时性要求,广义正交迭代算法的计算精度、鲁棒性和稳定性最好。特别在高噪声和远摄像距离条件下广义正交迭代算法更表现了其优越性能,而在该条件下最优绝对定向算法的角度误差太大,正交迭代算法的平移误差较大。

6 结 论

把正交迭代单目视觉算法扩展为正交迭代多目视觉算法,提出了广义正交迭代算法,是多摄像机系统位姿估计的通用算法,为多摄像机位姿估计提供了新的、精确的、鲁棒的、高效的解决方法。广义正交迭代算法的许多应用研究值得关注。目前,本课题组正构建一个基于广义正交迭代算法的航空器交会对接相对位姿光学测量系统。

参 考 文 献

1 Zhu Renzhan, Lin Yan, Zhang Lei. New algorithm of solving for ranges during final approach of spacecraft rendezvous[J]. *J. Beijing University of Aeronautics and Astronautics*, 2006, **7**(32): 764~768
朱仁璋,林彦,张磊. 航天器交会计算机视觉系统测距求解新

算法[J]. *北京航空航天大学学报*, 2006, **7**(32): 764~768
2 M. Fischler, R. C. Bolles. Random sample consensus: A paradigm for model fitting and automatic cartography[J]. *Comm. ACM*, 1981, **6**(24): 381~395
3 D. G. Lowe. Fitting parameterized three-dimensional models to images[J]. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1991, **13**(5): 441~450
4 C. P. Lu, G. Hager, E. Mjolsness. Fast and globally convergent pose estimation from video images[J]. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2000, **22**(5): 610~622
5 Adnan Ansar, Kostas Daniilidis. Linear pose estimation from points or lines[J]. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2003, **25**(5): 578~589
6 Ying Kin Yu, Kin Hong Wong, Michael Ming Yuen Chang. Pose estimation for augmented reality applications using genetic algorithm[J]. *IEEE Trans. on System, Man, and Cybernetics*, 2005, **36**(6): 1295~1301
7 Gerald Schweighofer, Axel Pinz. Robust pose estimation from a planar target[J]. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2006, **28**(12): 2024~2030
8 Cao Xibin, Zhang Shijie. An iterative method for vision-based relative pose parameters of RVD spacecrafts[J]. *J. Harbin Institute of Technology*, 2005, **37**(8): 691~694
曹喜滨,张世杰. 航天器交会对接位姿测量迭代算法[J]. *哈尔滨工业大学学报*, 2005, **37**(8): 691~694
9 Zhang Zhiyong, Zhang Jing, Zhu Dayong. A fast convergent pose estimation algorithm and experiments based on vision images[J]. *Acta Aeronautica et Astronautica Sinica*, 2007, **28**(4): 943~947
张志勇,张靖,朱大勇. 一种基于视觉成像的快速收敛的位姿测量算法及实验研究[J]. *航空学报*, 2007, **28**(4): 943~947
10 Mario L. Fravolini, Lorenzo Pollini, Brian Stolarik. A comparison of pose estimation algorithms for machine vision based aerial refueling for UAVs[C]. *Proceedings of the 2006 Conference on Control and Automation*, Mediterranean, June 2006. 1~6
11 C. Fermüller, Y. Aloimonos, P. Baker et al.. Multi-camera networks: eyes from eyes[C]. *Proc. IEEE Workshop on Omnidirectional Vision*, USA, 2000. 11~18
12 Abhinav Gupta, Anurag Mittal, Larry S. Davis. Constraint integration for efficient multiview pose estimation with self-occlusions[J]. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 2008, **30**(3): 493~506
13 Fredrik Viksten, Robert S. Oderberg, Klas Nordberg. Increasing pose estimation performance using multi-cue integration[C]. *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, Florida, May 2006. 3760~3767
14 B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternion[J]. *J. Opt. Soc. Am.*, 1987, **4**: 629~642
15 K. S. Arun, T. S. Huang, S. D. Blostein. Least-squares fitting of two 3-D point sets[J]. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1987, **9**(5): 698~100